CRYSTAL STRUCTURE OF CYTOCHROME P450

The present application is a continuation-in-part of applications PCT/GB02/02668 filed May 30, 2002 and designating the US, and Serial No. 10/221,036, filed April 2, 2002, and claims benefit of the following U.S. Provisional Application Serial Nos: 60/479,448, filed June 19, 2003; 60/421,063, filed October 25, 2002. US 10/221,036 claims the benefit of priority of 60/306,873, filed July 23, 2001, 60/306,874, filed July 23, 2001, and UK applications GB 0108214.8 filed April 2, 2001 and GB 0108212.2 filed April 2, 2001. The contents of all these applications are incorporated herein by reference.

Field of the Invention.

The present invention relates to the human cytochrome P450 protein 3A4, methods for its crystallization, crystals of 3A4 and their 3-dimensional structures, and uses thereof.

Background to the Invention.

Introduction to Cytochrome P450

Cytochrome P450s (CYP450) form a very large and complex gene superfamily of hemeproteins that metabolise physiologically important compounds in many species of microorganisms, plants and animals. Cytochrome P450s are important in the oxidative, peroxidative and reductive metabolism of numerous and diverse endogenous compounds such as steroids, bile, fatty acids, prostaglandins, leukotrienes, retinoids and lipids. Many of these enzymes also metabolise a wide range of xenobiotics including drugs, environmental compounds and pollutants. Their involvement in drug metabolism is extensive, it is estimated that 50% of all known drugs are affected in some way by the action of CYP450 enzymes. Significant resource is employed by the pharmaceutical industry to optimise drug candidates in order to avoid their detrimental interactions with the CYP450 enzymes. Another level of complication results from the fact that these enzymes exhibit different tissue distributions and polymorphisms between individuals and ethnic populations

Most mammalian P450s are located in the liver, but other organs and tissues have high concentrations of certain cytochrome P450s, including the intestinal wall, lung, kidney, adrenal cortex and nasal epithelium. Mammals have about 50 unique CYP450 genes and each family member is 45-55 KDa in size and contains a heme moiety that catalyses a two-electron activation of oxygen. The source of electrons may be used to classify CYP450s. Those that receive electrons in a three protein chain in which electrons flow from a flavin adenine dinucleotide (FAD) containing reductase, to an iron-sulphur protein, and then to P450 belong to the group of class I P450s, and include most of the bacterial enzymes. Class II P450s receive electrons from a reductase containing both FAD and flavin mononucleotide (FMN), and comprise the microsomal P450s that are the main culprits of drug metabolism. The mammalian microsomal cytochrome P450s are integral membrane proteins anchored by an N-terminal transmembrane spanning α-helix. They are inserted in the membrane of the endoplasmic reticulum by a short, highly hydrophobic N-terminal segment that acts as a non-cleavable signal sequence for insertion into the membrane. The remainder of the mammalian cytochrome P450 protein is a globular structure that protrudes into the cytoplasmic space. Hence, the bulk of the enzyme faces the cytoplasmic surface of the lipid bilayer. P450s require other membranous enzymatic components for activity including the flavoprotein NADPH-cytochrome P450 oxidoreductase and, in some cases, cytochrome b5. A single cytochrome P450 oxidoreductase supports the activity of all the mammalian microsomal enzymes by interacting directly with the P450s and transferring the required two electrons from NADPH. Cytochrome P450s are able to incorporate one of the two oxygen atoms of an O_2 molecule into a broad variety of substrates with concomitant reduction of the other oxygen atom by two electrons to H_2O. Cytochrome P450 are known to catalyse hydroxylations, epoxidation, N-, S-, and O-dealkylations, N-oxidations, sulfoxidations, dehalogenations, and other reactions.

The genes of the P450 superfamily have been categorized by Nelson *et al* (Pharmacogenetics, 6; 1-42, 1996) who proposed a systematic nomenclature for the family members. This nomenclature is used widely in the art, and is adopted herein. Nelson *et al* provide cross-references to sequence database entries for P450 sequences.

Homo sapiens has 17 cytochrome P450 gene families and 42 subfamilies that total more than 50 sequenced isoforms. Cytochrome P450s from families 1, 2 and 3 constitute the major pathways for drug metabolism. Many drugs rely on hepatic metabolism by cytochrome P450s for clearance from the circulation and for pharmacological inactivation. Conversely, some drugs have to be converted in the body to their pharmacologically active metabolites by P450s. Many promising lead compounds are terminated in the development phase due to their interaction with one or more P450s. One of the greatest problems in drug discovery is the prediction of the role of cytochrome P450s on the metabolism or modification of drug leads. Early detection of metabolic problems associated with a chemical lead series is of paramount importance for the pharmaceutical industry. Obtaining crystal structures of the main human drug metabolising cytochrome P450s would be highly valuable for drug design, as this would provide detailed information on how P450 enzymes recognize drug molecules and the mode of drug binding. This in turn would allow drug companies to develop strategies to modify metabolic clearance and decrease the attrition rates of compounds in development.

The major human CYP450 isoforms involved in drug metabolism are CYP1A2, CYP2C9, CYP2C19, CYP2D6 and CYP3A4. The level of sequence identity between these family members ranges from about 20-80%, with much of the variability within the residues involved in substrate recognition. CYP450 enzymes are also present in bacteria and much of the understanding of substrate recognition is derived from crystal structures obtained of bacterial CYP450 enzymes.

CYP3A is both the most abundant and most clinically significant subfamily of cytochrome P450 enzymes. The CYP3A subfamily has four human isoforms, 3A4, 3A5, 3A7 and 3A43, CYP3A4 being the most commonly associated with drug interactions. The CYP3A isoforms make up approximately 50% of the liver's total cytochrome P450 and are widely expressed throughout the gastrointestinal tract, kidneys and lungs and therefore are ultimately responsible for the majority of first-pass metabolism. This is important as increases or decreases in first-pass metabolism can have the effect of administering a much smaller or larger dose of drug than usual. More than 150 drugs are known substrates of CYP3A4, including many of the opiate analgesics, steroids, antiarrhythmic agents, tricyclic antidepressants, calcium-channel blockers and macrolide antibiotics. Although several substrates show age-dependent reductions in elimination, the enzyme itself does not appear to be altered. CYP3A4 is important in the metabolism of many drugs including cyclosporine, codeine, tamoxifen, lovastatin, and many more, and endogenous compounds such as testosterone, estradiol and cortisol. Ketoconazole, itraconazole, erythromycin, clarithromycin, diltiazem, fluvoxamine, nefazodone, and dihydroxybergamottin and various substances found in grapefruit juice, green tea and other foods are potent inhibitors of CYP3A4 and are known to be responsible for many drug interactions. These interactions can have serious clinical consequences.

Background to Crystallisation

It is well-known in the art of protein chemistry, that crystallising a protein is a chancy and difficult process without any clear expectation of success. It is now evident that protein crystallization is the main hurdle in protein structure determination. For this reason, protein crystallization has become a research subject in and of itself, and is not simply an extension of the protein crystallographer's laboratory. There are many references which describe the difficulties associated with growing protein crystals. For example, Kierzek, A.M. and Zielenkiewicz, P., (2001), Biophysical Chemistry, 91, 1-20, *Models of protein crystal* growth, and Wiencek, J.M. (1999) Annu. Rev. Biomed. Eng., 1, 505-534, *New Strategies for crystal growth*.

It is commonly held that crystallization of protein molecules from solution is the major obstacle in the process of determining protein structures. The reasons for this are many; proteins are complex molecules, and the delicate balance involving specific and non-specific interactions with other protein molecules and small molecules in solution, is difficult to predict.

Each protein crystallizes under a unique set of conditions, which cannot be predicted in advance. Simply supersaturating the protein to bring it out of solution may not work, the result would, in most cases, be an amorphous precipitate. Many precipitating agents are used, common ones are different salts, and polyethylene glycols, but others are known. In addition, additives such as metals and detergents can be added to modulate the behaviour of the protein in solution. Many kits are available (e.g. from Hampton Research), which attempt to cover as many parameters in crystallization space as possible, but in many cases these are just a starting point to optimise crystalline precipitates and crystals which are unsuitable for diffraction analysis. Successful crystallization is aided by a knowledge of the proteins behaviour in terms of solubility, dependence on metal ions for correct folding or activity, interactions with other molecules and any other information that is available. Even so, crystallization of proteins is often regarded as a time-consuming process, whereby subsequent experiments build on observations of past trials.

In cases where protein crystals are obtained, these are not necessarily always suitable for diffraction analysis; they may be limited in resolution, and it may subsequently be difficult to improve them to the point at which they will diffract to the resolution required for analysis. Limited resolution in a crystal can be due to several things. It may be due to intrinsic mobility of the protein within the crystal, which can be difficult to overcome, even with other crystal forms. It may be due to high solvent content within the crystal, which consequently results in weak scattering. Alternatively, it could be due to defects within the crystal lattice which mean that the diffracted x-rays will not be completely in phase from unit to unit within the lattice. Any one of these or a combination of these could mean that the crystals are not suitable for structure determination.

Some proteins never crystallize, and after a reasonable attempt it is necessary to examine the protein itself and consider whether it is possible to make individual domains, different N or C-terminal truncations, or point mutations. It is often hard to predict how a protein could be re-engineered in such a manner as to improve crystallisability. Our understanding of crystallisation mechanisms are still incomplete and the factors of protein structure which are involved in crystallisation are poorly understood.

Determination of protein structure.

A mathematical operation termed a Fourier transform relates the diffraction pattern observed from a crystal and the molecular structure of the protein comprising the crystal. A Fourier transform may be considered to be a summation of sine and cosine waves each with a defined amplitude and phase. Thus, in theory, it is possible to calculate the electron density associated with a protein structure by carrying out an inverse Fourier transform on the diffraction data. This, however, requires amplitude and phase information to be extracted from the diffraction data. Amplitude information may be obtained by analysing the intensities of the spots within a diffraction pattern. The conventional methods for recording diffraction data do, however, mean that any phase information is lost. This "phase information" must be in some way recovered and the loss of this information represents the "crystallographic phase problem". The phase information necessary for carrying out the inverse Fourier transform can be obtained via a variety of methods. If a protein structure exists a set of theoretical amplitudes and phases may be calculated using the protein model and then the theoretical phases combined with the experimentally derived amplitudes. An electron density map may then be calculated and the protein structure observed.

If there is no known structure of the protein then alternative methods for obtaining phases must be explored. One method is multiple isomorphous replacement (MIR). This relies on soaking "heavy atom" (*i.e.* platinum, uranium, mercury, *etc*) compounds into the crystals and observing how their incorporation into the crystals modifies the spot intensities observed in the diffraction pattern. This method relies on the heavy atoms being incorporated into the protein at a finite number of defined sites. It is a pre-requisite of an isomorphous replacement experiment that the heavy atom soaked crystals remain isomorphous. That is, there should be no appreciable alterations in the physical characteristics of the protein crystal (*i.e.* perturbations to crystallographic cell dimensions, or significant loss of resolution). Perturbations to the physical properties of the crystal are termed non-isomorphisms and prevent this type of experiment being successfully completed. Successful isomorphous incorporation of heavy atoms into a protein crystal results in the intensities of the spots within the diffraction pattern obtained from the crystal being modified, as compared to the data collected from an identical, unsoaked, (native) crystal. The diffraction data obtained from a successful isomorphous replacement experiment are termed a "derivative" dataset. By mathematically analysing the "native" and "derivative" datasets it is possible to extract preliminary phase information from the datasets. This phase information, when combined with the experimentally obtained amplitudes from the native dataset, enables an electron density map of the unknown protein molecule to be calculated using the Fourier transform method.

An alternative method for obtaining phase information for a protein of unknown structure is to perform a multi-wavelength anomalous dispersion (MAD) experiment. This relies on the absorption of X-rays by electrons at certain characteristic X-ray wavelengths. Different elements have different characteristic absorption edges. Anomalous scattering by atoms within a protein will modify the diffraction pattern obtained from the protein crystal. Thus if a protein contains atoms which are capable of anomalous scattering a diffraction dataset (anomalous dataset) may be collected at an X-ray wavelength at which this anomalous scattering is maximal. By altering the X-ray wavelength to a value at which there is no anomalous scattering a native dataset may then be collected. Similarly to the MIR case, by mathematically processing the anomalous and native datasets the phase information necessary for the calculation of an electron density map may be determined. The most usual way to introduce anomalous scatterers into a protein is to replace the sulphur containing methionine amino acid residues with selenium containing seleno-methionine residues. This is done by generating recombinant protein that is isolated from cells grown on growth media that contain seleno-methionine. Selenium is capable of anomalously scattering X-rays and may thus be used for a MAD experiment. Further methods for phase determination such as single isomorphous replacement (SIR), single isomorphous replacement anomalous scattering (SIRAS) and direct methods exist, but the principles behind them are similar to MIR and MAD.

The final method generally available for the calculation of the phases necessary for the determination of an unknown protein structure is molecular replacement. This method relies upon the assumption that proteins with similar amino acid sequences (primary sequences) will have a similar fold and three-dimensional structure (tertiary structure). Proteins related by amino acid sequence are termed homologous proteins. If an X-ray diffraction dataset has been collected from a crystal whose protein structure is not known, but a structure has been determined for a homologous protein, then molecular replacement can be attempted. Molecular replacement is a mathematical process that attempts to correlate the dataset obtained from a new protein crystal with the theoretical diffraction pattern calculated for a protein of known structure. If the correlation is sufficiently high some phase information can be extracted from the known protein structure and combined with the amplitudes obtained from the new protein dataset. This enables calculation of a preliminary electron density map for the protein of unknown structure.

If an electron density map has been calculated for a protein of unknown structure then the amino acids comprising the protein must be fitted into the electron density for the protein. This is normally done manually, although high resolution data may enable automatic model building. The process of model building and fitting the amino acids to the electron density can be both a time consuming and laborious process. Once the amino acids have been fitted to the electron density it is necessary to refine the structure. Refinement attempts to maximise the correlation between the experimentally calculated electron density and the electron density calculated from the protein model built. Refinement also attempts to optimise the geometry and disposition of the atoms and amino acids within the user-constructed model of the protein structure. Sometimes manual re-building of the structure will be required to release the structure from local energetic minima. There are now several software packages available that enable an experimentalist to carry out refinement of a protein structure. There are certain geometry and correlation diagnostics that are used to monitor the progress of a refinement. These diagnostic parameters are monitored and rebuilding/refinement continued until the experimenter is satisfied that the structure has been adequately refined.

Description of anomalous scattering theory

If the energy of incident X-rays is close to the minimum energy that is required to eject a bound electron from an innermost shell of an atom, the scattering of the X-rays is described as "anomalous". In the process of "normal" scattering, the electrons are forced to undergo vibrations at the same frequency as that of the incident X-ray photon, emitting elastically scattered photons (i.e. no change in frequency) in the process. However, because this frequency is far from the natural frequency of vibration of the electron there is no effect on the scattered photon from this natural vibration. In the process of "anomalous" scattering, the frequency of the incident photon is close to the natural frequency of the electron, resulting in a resonance effect, which is manifested as a dispersion (decrease in velocity, though still no change in frequency) of the photon, as well as a vibration damping effect, which is manifested as absorption (decrease in intensity) of a fraction of the incident photons.

The anomalously scattered photon will thus have a phase angle associated with it that is retarded when compared with one being scattered normally, all other conditions being equal. If the structure consists of a mixture normal and anomalous scatterers this phase lag results in the breakdown of Friedel's law, as pairs of reflections with indices (h,k,l) and (-h,-k,-l) that are diffracted from opposite sides of the same crystal plane no longer have the same amplitudes.

By careful measurement of the two reflection intensities, and by consideration of their relative amplitudes, it is possible to make an initial estimate of the phases of all reflections that have been observed.

In theory all atoms could give rise to an anomalous scattering effect if irradiated with X-ray radiation of the appropriate wavelength. However as the scattering is directly proportional to the weight of the scatterer, heavier elements are normally chosen, e.g. sulphur or larger. The choice of element is also dependent on the ability to tune the energy of the X-rays to the required transition energy. As access to tuneable synchrotron X-radiation has become routine, the MAD technique has come of age. Incorporation of an anomalous scatterer may be via a number of routes e.g. by soaking crystals in solutions containing heavy atoms which then bind to the protein, by expressing recombinant proteins in media in which an element has been replaced by a suitable heavier element (e.g. the replacement of methionine with selenomethionine) leading to the incorporation of the element in certain amino acids themselves, or making use of naturally occurring co-factors which contain heavy elements.

As the contribution from the anomalous scatterer may be small, it is often important to obtain well-recorded, redundant data, and to facilitate detection of what may be a small signal, it is helpful to have a reference dataset to which the anomalous dataset can be compared. The routine collection of X-ray data at cryo-temperatures has prolonged crystal lifetime and has made collection of multiple datasets (at different wavelengths) from a single crystal now feasible for many crystal systems. Collection and analysis of multiple datasets from a single crystal has the advantage of eliminating all effects related to non-isomorphism (variations in structure between different crystals due to random variations in soaking and/or freezing conditions).

In the case of cytochrome P450, the haem group that forms the site of enzymatic activity naturally contains a single iron atom. Iron has transition energies at the high energies (long wavelengths) obtainable at tunable synchrotron beamlines.

P450 Crystal Stuctures.

As of 2002, eight cytochrome P450 structures had been solved by X-ray crystallography and were available in the public domain. All of the cytochrome P450s, whose structures had been solved, were expressed in *E. coli.* Six structures correspond to bacterial cytochrome P450s: P450cam (CYP101 Poulos *et al.*, 1985, *J. Biol. Chem.*, 260, 16122), the hemeprotein domain of P450BM3 (CYP102, Ravichandran *et al.*, 1993, *Science*, 261, 731), P450terp (CYP108, Hasemann *et al.*, 1994, *J. Mol. Biol.* 236, 1169), P450eryF (CYP107A1, Cupp-Vickery and Poulos, 1995, *Nature* Struct. Biol. 2, 144), P450 14α-sterol demethylase (CYP51, Podust *et al.*, 2001, *Proc. Natl. Acad. Sci. USA*, 98, 3068) and the crystal structure of a thermophilic cytochrome P450 (CYP119) from Archaeon sulfolobus solfataricus was solved (Yano *et al.*, 2000, *J. Biol. Chem.* 275, 31086). The structure of cytochrome P450nor was obtained from the denitrifying fungus Fusarium oxysporum (Shimizu *et al.* 2000, *J. Inorg. Biochem.* 81, 191). The eighth structure is that of the rabbit 2C5 isoform, the first structure of a mammalian cytochrome P450 (Williams *et al.* 2000, *Mol. Cell.* 5, 121).

WO 03/035693 describes the crystallisation of a human 2C9 P450 protein molecule and provides an analysis of the protein crystal structure.

The reason why the mammalian cytochrome P450s have been particularly difficult to crystallize, compared to their bacterial counterparts, resides in the nature of these proteins. The bacterial cytochrome P450s are soluble whereas the mammalian P450s are membrane-associated proteins. Thus, structural studies on mammalian cytochrome P450s may use the combination of heterologous expression systems that allow expression of single cytochrome P450s at high concentration with modification of their sequences to improve the solubility and the behaviour of these proteins in solution.

Due to significant sequence differences from both the bacterial proteins and rabbit proteins, to fully understand the role of the human CYP450 enzymes in drug metabolism, the crystal structures of other human isoforms are still required.

Disclosure of the Invention.

The present invention relates to the crystal structure of human 3A4, which allows the binding location of the substrates in the enzyme to be investigated and determined.

More particularly, the present inventors have obtained an electron density map for 3A4 which is useful for the provision of atomic coordinate models of this protein, and also for other applications which are discussed in Section H below. In addition, the data of Table 3 herein provides structure factor phase data, permitting others of skill in the art to solve X-ray diffraction data of 3A4 and homologous protein crystals more readily in order to provide electron density maps.

In a further aspect, the invention provides a three dimensional structure of 3A4 set out in Table 5, and uses thereof.

In general aspects, the present invention is concerned with the provision of a 3A4 structure and its use in modelling the interaction of molecular structures, e.g. potential and existing pharmaceutical compounds, prodrugs, P450 inhibitors or substrates, or fragments of such compounds, prodrugs, inhibitors or substrates with this 3A4 structure.

These and other aspects and embodiments of the present invention are discussed below.

The above aspects of the invention, both singly and in combination, all contribute to features of the invention, which are advantageous.

Brief Description of the Tables

Table 1 provides the data statistics
Table 2 provides the phasing statistics.
Table 3 (Figure 1) provides the structure factors and phases which can be used to generate an electron density map of the 3A4 crystal structure.
Table 4 provides refinement statistics.
Table 5 (Figure 2) sets out the coordinate data of the structure of 3A4.
Table 6 (Figure 3) sets out one possible set of coordinate data of a loop region of 3A4.
Table 7 details binding site residues of 3A4.
Table 8 sets out newly identified binding site residues of 3A4.

Brief Description of the Drawings

Figure 1 sets out Table 3.
Figure 2 sets out Table 5.
Figure 3 sets out Table 6.

Detailed Description of the Invention

A. Protein Crystals.

The present invention provides a crystal of 3A4 having an orthorhomobic space group I222, and unit cell dimensions 78 Å, 100 Å, 132 Å, 90°, 90°, 90°. Unit cell variability of 5% may be observed in all dimensions.

Such a crystal may be obtained using the methods described in the accompanying examples.

The crystal may be of a 3A4 protein which is desirably truncated in its N-terminal region to delete the hydrophobic trans-membrane domain, and the region is replaced by a short (e.g. 8 to 20) amino acid sequence. For expression of the human 3A4 P450, we have used an N-terminal sequence MAYGTHSHGLFKKLGI in place of the native N-terminal residues, which increases expression of the proteins in *E. coli* and increases solubility.

The 3A4 P450 may optionally comprise a tag, such as a C-terminal polyhistidine tag to allow for recovery and purification of the protein.

Our experiments have been based on the use of the particular N-terminal truncation mentioned above, and this protein also comprises a polyhistidine tag at the C-terminus. The N-terminal truncation and tag are both features which can be varied by those of skill in the art using routine skill. For example, alternative N-terminal sequence might be utilised, for example for production in host cells other than *E. coli.* Likewise, other tags may be used for purification of the protein as described below. These N- and C-terminal modification may be made to a 3A4 protein which retains the core sequence of the wild type protein from the residue 17 onwards of SEQ ID NO:2 shown herein, up to the residue immediately preceding the polyhistidine tag.

Where present, the N-terminal sequence is preferably not the full length wild-type sequence, and preferably smaller than 30, e.g. 20 residues in size. Preferably, it is shorter that the wild type sequence. Preferably, the N-terminal region is the truncation illustrated in the accompanying examples. This type of N-terminal sequence reduces the tendency of 3A4 to anchor to membranes and to aggregate compared to the wild type sequence. The truncation utilised here has wild-type residues 3-24 deleted.

Where present, the C-terminal sequence is preferably no larger than 30, and preferably no larger than 10 amino acids in size.

The 3A4 sequence may be that of the core sequence illustrated herein, or an allele thereof, or a variant which retains the ability to form crystals under the conditions illustrated herein. Such variants include those with a number of amino acid substitutions, for example 1, 2, 3, 4, 5, 6, 7, 8, 9 or 10 amino acids by an equivalent or fewer number of amino acids. Further examples of variants, including mutants, are discussed further herein below.

The methodology used to provide a P450 crystal illustrated herein may be used generally to provide a human 3A4 crystal resolvable at a resolution of at least 3.0 Å, and preferably at least 2.8 Å.

The invention thus further provides a 3A4 crystal having a resolution of at least 3.0 Å, preferably at least 2.8 Å.

The proteins may be wild-type proteins or variants thereof, which are modified to promote crystal formation, for example by N-terminal truncations and/or deletion of loop regions, which prevent crystal formation.

In a further aspect, the invention provides a method for making a P450 protein crystal, particularly of a 3A4 protein comprising the core sequence of 3A4 (as defined above) or a variant thereof, which method comprises growing a crystal by vapor diffusion using a reservoir buffer that contains 0.05-0.2 M HEPES pH 7.0-7.8, 2.5-10% IPA, 0-20% PEG 4000, 0-0.3 M sodium chloride, 0-10% PEG 400, 0-10% glycerol, preferably 0.1 M HEPES pH 7.2, 5% IPA, 10% PEG 4000. The crystal is grown by vapor diffusion and is performed by placing an aliquot of the solution on a cover slip as a hanging drop above a well containing the reservoir buffer. The concentration of the protein solution used was 0.3-0.7 mM.

Crystals of the invention also include crystals of 3A4 mutants, chimeras, homologues in the 3A family (e.g. 3A1, 3A5, 3A7, 3A12 and 3A43) and alleles.

(i) Mutants

A mutant is a 3A4 protein characterized by the replacement or deletion of at least one amino acid from the wild type 3A4. Such a mutant may be prepared for example by site-specific mutagenesis, or incorporation of natural or unnatural amino acids.

The present invention contemplates "mutants" wherein a "mutant" refers to a polypeptide which is obtained by replacing at least one amino acid residue in a native or synthetic 3A4 with a different amino acid residue and/or by adding and/or deleting amino acid residues within the native polypeptide or at the N- and/or C-terminus of a polypeptide corresponding to 3A4, and which has substantially the same three-dimensional structure as 3A4 from which it is derived. By having substantially the same three-dimensional structure is meant having a set of atomic structure co-ordinates that have a root mean square deviation (r.m.s.d.) of less than or equal to about 2.0 Å (preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å) when superimposed with the atomic structure co-ordinates of the

3A4 from which the mutant is derived when at least about 50% to 100% of the C_α atoms of the 3A4 are included in the superposition. A mutant may have, but need not have, enzymatic or catalytic activity.

To produce homologues or mutants, amino acids present in the said protein can be replaced by other amino acids having similar properties, for example hydrophobicity, hydrophobic moment, antigenicity, propensity to form or break α-helical or β-sheet structures, and so on. Substitutional variants of a protein are those in which at least one amino acid in the protein sequence has been removed and a different residue inserted in its place. Amino acid substitutions are typically of single residues but may be clustered depending on functional constraints e.g. at a crystal contact. Preferably amino acid substitutions will comprise conservative amino acid substitutions. Insertional amino acid variants are those in which one or more amino acids are introduced. This can be amino-terminal and/or carboxy-terminal fusion as well as intrasequence. Examples of amino-terminal and/or carboxy-terminal fusions are affinity tags, MBP tag, and epitope tags.

Amino acid substitutions, deletions and additions which do not significantly interfere with the three-dimensional structure of the 3A4 will depend, in part, on the region of the 3A4 where the substitution, addition or deletion occurs. In highly variable regions of the molecule, non-conservative substitutions as well as conservative substitutions may be tolerated without significantly disrupting the three-dimensional structure of the molecule. In highly conserved regions, or regions containing significant secondary structure, conservative amino acid substitutions are preferred.

Conservative amino acid substitutions are well-known in the art, and include substitutions made on the basis of similarity in polarity, charge, solubility, hydrophobicity, hydrophilicity and/or the amphipathic nature of the amino acid residues involved. For example, negatively charged amino acids include aspartic acid and glutamic acid; positively charged amino acids include lysine and arginine; amino acids with uncharged polar head groups having similar hydrophilicity values include the following: leucine, isoleucine, valine; glycine, alanine; asparagine, glutamine; serine, threonine; phenylalanine, tyrosine. Other conservative amino acid substitutions are well known in the art.

In some instances, it may be particularly advantageous or convenient to substitute, delete and/or add amino acid residues in order to provide convenient cloning sites in the cDNA encoding the polypeptide, to aid in purification of the polypeptide, etc. Such substitutions, deletions and/or additions which do not substantially alter the three dimensional structure of 3A4 will be apparent to those having skills in the art.

It should be noted that the mutants contemplated herein need not exhibit enzymatic activity. Indeed, amino acid substitutions, additions or deletions that interfere with the catalytic activity of the 3A4 but which do not significantly alter the three-dimensional structure of the catalytic region are specifically contemplated by the invention. Such crystalline polypeptides, or the atomic structure co-ordinates obtained there from, can be used to identify compounds that bind to the protein.

The residues for mutation could easily be identified by those skilled in the art and these mutations can be introduced by site-directed mutagenesis e.g. using a Stratagene QuikChange^{TM} Site-Directed Mutagenesis Kit or cassette mutagenesis methods (see e.g. Ausubel et al., eds., *Current Protocols in Molecular Biology*, John Wiley & Sons, Inc., New York, and Sambrook et al., *Molecular Cloning: a Laboratory Manual*, 2nd ed., Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY, (1989)).

(ii) Alleles

The present invention contemplates "alleles" wherein allele is a term coined by Bateson and Saunders (1902) for characters which are alternative to one another in Mendelian inheritance (Gk. Allelon, one another; morphe, form). Now the term allele is used for two or more alternative forms of a gene resulting in different gene products and thus different phenotypes. An allele contains nucleotide changes that have been shown to affect transcription, splicing, translation, post-transcriptional or post-translational modifications or result in at least one amino acid change. These different alleles are particularly important in P450s as some confer different metabolic clearance rates of specific drugs onto the phenotype. Alleles of P450s are often only different by one or two amino acids. As of 2002, 25 alleles of 3A4 have been identified, where wild type is CYP3A4*1A (NCBI ACCESSION M18907, Gonzalez FJ, Schmid BJ, Umeno M, Mcbride OW, Hardwick JP, Meyer UA, Gelboin HV, Idle JR, DNA 1988 Mar;7(2):79-86).

To the extent that the present invention relates to 3A4-ligand complexes and mutant, homologue, analogue, allelic form, species variant proteins of 3A4, crystals of such proteins may be formed. The skilled person would recognize that the conditions provided herein for crystallising 3A4 may be used to form such crystals. Alternatively, the skilled person would use the conditions as a basis for identifying modified conditions for forming the crystals.

Thus the aspects of the invention relating to crystals of 3A4, may be extended to crystals of mutant and mutants of 3A4 which result in homologue, allelic form, and species variant.

(iii) Crystallization of 3A4

To produce crystals of 3A4 protein the final protein is, conveniently, concentrated to 10-60, e.g. 20-40 mg/ml in 10-100 mM potassium phosphate with high salt (e.g. 500 mM NaCl or KCl) by using concentration devices which are commercially available. Crystallisation of the protein is set up by the 0.5-2 μl hanging drop method and the protein is crystallised by vapour diffusion at 5-25 °C against a range of vapour diffusion buffer compositions.

Typically the vapour diffusion buffer comprises 0 – 27.5%, preferably 2.5-27.5% PEG 1K-20 K, preferably 1-8K or PEG 2000MME-5000MME, preferably PEG 2000 MME, or 0-10% Jeffamine M-600 and/or 5-20%, e.g. 10-20% propanol or 15-20% ethanol or about 15%-30%, e.g. about 15% 2-methyl-2,4-pentanediol (MPD), optionally with 0.01 M –1.6 M salt or salts and/or 0-0.15, e.g. 0-0.1, M of a solution buffer and/or 0-35%, such as 0-15%, glycerol and/or 0-35% PEG300-400; but preferably:

10-25% PEG 1K-8K or PEG 2000MME or 0-10% Jeffamine M-600 and/or 5-15%, e.g. 10-15%, propanol or ethanol, optionally with 0.1 M –0.2 M salt or salts and/or 0-0.15, e.g. 0-0.1 M solution buffer and/or PEG400, but more preferably:

15-20% PEG 3350 or PEG 4000 or PEG 2000MME or 0-10% Jeffamine M-600 or 5-15%, e.g. 10-15% propanol or ethanol, optionally with 0.1 M –0.2 M salt or salts and/or 0-0.15 M solution buffer.

The salt may be an alkali metal (particularly lithium, sodium and potassium), alkaline earth metal (e.g. magnesium or calcium), ammonium, ferric, ferrous or transition metal salt (e.g. zinc) of a halide (e.g. bromide, chloride or fluoride), acetate, formate, nitrate, sulfate, tartrate, citrate or phosphate. This includes sodium fluoride, potassium fluoride, ammonium fluoride, ammonium acetate, lithium acetate, magnesium acetate, sodium acetate, potassium acetate, calcium acetate, zinc acetate, ammonium chloride, lithium chloride, magnesium chloride, potassium chloride, sodium chloride, potassium bromide, magnesium formate, sodium formate, potassium formate, ammonium formate, ammonium nitrate, lithium nitrate, potassium nitrate, sodium nitrate, ammonium sulfate , potassium sulfate, lithium sulfate, sodium sulfate, di-sodium tartrate, potassium sodium tartrate, di-ammonium tartrate, potassium dihydrogen phosphate, tri-sodium citrate, tri-potassium citrate, zinc acetate, ferric chloride, calcium chloride, magnesium nitrate, magnesium sulfate, sodium dihydrogen phosphate, di-sodium hydrogen phosphate, di-potassium hydrogen phosphate, ammonium dihydrogen phosphate, di-ammonium hydrogen phosphate, tri-lithium citrate, nickel chloride, ammonium iodide, di-ammonium hydrogen citrate.

Solution buffers if present include, for example, Hepes, Tris, imidazole, cacodylate, tri-sodium citrate/citric acid, tri-sodium citrate/HCl, acetic acid/sodium acetate, phosphate-citrate, sodium potassium phosphate, 2-(N-morpholino)-ethane sulphonic acid/NaOH (MES), CHES or bis-trispropane.

The pH range is desirably maintained at pH 4.2-8.5, preferably 4.7-8.5.

Solution buffers if present can also include, for example, bicine, bis-tris, CAPS, MOPS, ADA which allow the pH to be maintained in the range 5.8-11.

Crystals may be prepared using a Hampton Research Screening kits, Poly-ethylene glycol (PEG)/ion screens, PEG grid, Ammonium sulphate grid, PEG/ammonium sulphate grid or the like.

Crystallisation may also be performed in the presence of an inhibitor of P450, e.g. fluoroxamine or 2-phenyl imidazole. 3A4 crystallisation may also be performed in the presence of one or more inhibitors e.g. ketoconazole and/or in the presence of one or more substrate(s) e.g. testosterone.

Additives can be added to a crystallisation condition identified to influence crystallisation. Additive Screens are to be used during the optimisation of preliminary crystallisation conditions where the presence of additives may assist in the crystallisation of the sample and the additives may improve the quality of the crystal e.g. Hampton Research additive screens which use glycerol, polyols and other protein stabilizing agents in protein crystallisation (R. Sousa. Acta. Cryst. (1995) D51, 271-277) or divalent cations (Trakhanov, S. and Quiocho, F.A. Protein Science (1995) 4,9, 1914-1919).

In addition, detergents may be added to a crystallisation condition to improve the crystallisation behaviour e.g. the ionic, non-ionic and zwitterionic detergents found in the Hampton Research detergent screens (McPherson, A., et al., The effects of neutral detergents on the crystallization of soluble proteins, J. Crystal Growth (1986) 76, 547-553).

Alternatively, the vapour diffusion buffer typically comprises 0 – 27.5% PEG 1K-20 K, preferably 1-8K or PEG 2000MME-5000MME, preferably PEG 2000 MME, or 0-10% Jeffamine M-600 and/or 1-20%, e.g. 1-20% propanol or 15-20% ethanol or about 1%-30%, e.g. about 2-25% 2-methyl-2,4-pentanediol (MPD), optionally with 0.01 M –1.6 M salt or salts and/or 0-0.15 M, e.g. 0-0.1 M, of a solution buffer and/or 0-35%, such as 0-15%, glycerol and/or 0-35% PEG300-400; but preferably:

0 – 27.5%, preferably 2.5-27.5% PEG 1K-20 K, most preferably 5-20% PEG 4K or PEG 2000MME-5000MME, preferably PEG 2000 MME, and 1-20% alcohol, e.g. 1-20% propanol e.g. iso-propanol or 2-25% 2-methyl-2,4-pentanediol (MPD), optionally with 0.01 M –1.6 M salt or salts and/or 0-0.15 M, e.g. 0-0.1 M, of a solution buffer and/or 0-35%, such as 0-15%, glycerol and/or 0-35% PEG300-400.

B. Electron Density Map

In one aspect, the invention provides a crystal of 3A4 having the structure factors and phases of Table 3.

In a further aspect, the invention also provides a crystal of P450 having the electron density map generated from the data of Table 3.

An advantageous feature of the electron density map is that it has a resolution of about 2.8 Å.

Table 3 has eight columns. The first three columns are the indices h, k and l of each individual reflection. Columns four and five are the experimentally measured structure factors and the associated standard deviations of the peak wavelength, respectively. Column six is the solvent flattened structure factor amplitude. Column seven is the solvent flattened structure factor phase. Column eight is the solvent flattened figure of merit associated with the reflection. The data of columns six to eight were generated from the experimentally measured structure factors and by using the phasing procedure in SHARP (see equation (2) in de la Fortelle & Bricogne, 1997) that are then used in density modification.

The best electron density map for structural interpretation is then calculated via a Fourier transform, using the following formula

$$\rho(x, y, z) = \frac{1}{V} \sum_{h} \sum_{k} \sum_{l} |F(h, k, l)| \exp[-2\pi i(hx + ky + lz) + i\varphi(h, k, l)]$$

Thus the electron density map can be generated from Table 3 using columns six and seven using, for example, the FFT program which is part of the CCP4 suite of programs (Collaborative Computational Project 4. The CCP4 Suite: Programs for Protein Crystallography, *Acta Crystallographica*, D50, (1994), 760-763.). The resulting electron density map can then be viewed, interpreted or models built into it using a crystallographic graphical viewing program such as “O” (Jones et al., *Acta Crystallographica*, A47, (1991), 110-119) or “QUANTA” (1994, San Diego, CA: Molecular Simulations, Jones et al., Acta Crystallography A47 (1991), 110-119).

Errors in electron density maps derive principally from errors in the phase angles (ϕ) of the structure factors used in their calculation; errors in the corresponding amplitudes ($|F|$) are normally insignificant in comparison. The expected error in the phase of a structure factor is normally expressed as a “figure of merit” (m), which can be defined as the expected value of the cosine of the error in the “best” phase (that value of the phase which minimises the root-mean-square error $\sigma(\rho)$ in the electron density).

$$m = \langle \cos(\Delta\phi_{best}) \rangle$$

The actual (but unknown) phase error will vary significantly from one structure factor to the next, partly because of the random nature of experimental error and partly because structure factors with small amplitudes on average tend to have larger errors than those with large amplitudes (small amplitudes clearly do not contribute to the electron density summation as much as large ones; a structure factor with zero amplitude contributes nothing and so has a phase angle which is completely indeterminate). In addition, the phase error will tend to be greater at high resolution, because, for example, the small errors in locating the atoms used in the phase calculation have a greater effect at high resolution. For these reasons the figures of merit and phase errors are normally binned together and averaged either according to amplitude or to resolution (we have chosen to present the averaged figures of merit by resolution).

Blow & Crick (Blow, D. M. and Crick, F. H. C. Acta Cryst. (1959) 12, 794-802) derived an estimate of the RMS error in the electron density:

$$\sigma(\rho) = V^{-1} (\Sigma_h\Sigma_k\Sigma_l \,|F_{hkl}|^2 (1-m_{hkl}{}^2))^{½}$$

where the summation must be performed over the entire sphere of reciprocal space, not just the asymmetric unit.

Taking this formula, the above definition of the figure of merit and the above argument concerning the dependence of the phase error on the amplitude, it is suggested that for future purposes of comparing any set of phases with those in Table 3 the following weighted average of the cosine of the phase difference should be calculated:

$$\cos(\Delta\phi_{mean}) = (\Sigma_h\Sigma_k\Sigma_l \,|F_{hkl}|^2 \cos^2(\Delta\phi) \,/\, \Sigma_h\Sigma_k\Sigma_l \,|F_{hkl}|^2)^{½}$$

where the summations are performed in resolution shells, as well as over the entire sphere. From this the average phase difference $\Delta\phi_{mean}$ for the shell can be obtained: this is a measure of the average similarity of the two sets of phases, which may then be directly compared with the expected values of the phase error in column 10 of Table 2. Thus a value of the average phase difference less than the expected phase error for most of the resolution shells would imply that the two phase sets are providing similar information.

From Table 2 it can be seen that the average phase error for the phases in Table 3 is 45°, and hence if the difference between a second set of phases and the set of phases in Table 3 is less than 45° (over the same resolution) for the purposes of this invention the two set of phases and there resulting maps will be considered to be equivalent. The skilled person would understand that the values of the phases would change for a different origin of the coordinates and would make the appropriate adjustments.

This electron density map will allow the placement of a large percentage of all the atoms of 3A4, and reveals for the first time the spatial arrangement of the atoms of 3A4. Knowledge of the spatial arrangement of these atoms has clear implications in various fields. For example, knowledge of those atoms that form the enzymatic active site of the molecule will determine the physico-chemical properties of compounds that are ligands for the enzyme. The ability to modify these properties and hence to ultimately modify the enzyme's ability to metabolise a particular compound has clear value to the pharmacological industry. An indicator of the quality of the phases used to generate the map is as follows: inspection of anomalous log-likelihood gradient maps within SHARP (La Fortelle, E. de and Bricogne, G. (1997) *Methods in Enzymology* **276,** 472-494) using the current heavy atom model reveals several peaks that correlate with the position of the sulphur atoms from cysteine and methionine residues (there is an expected contribution to the anomalous scattering from sulphur atoms of cysteine and methionine residues within the protein at the long wavelengths used to collect the data). The identification of the location of sulphur containing residues will facilitate assignment of the protein sequence to the model that will be built into the electron density map.

The data of Table 3 will in practice be used by those of skill in the art in electronic form to allow for processing of the data by computer programs such as those discussed herein. Thus in practice the programs will use all the data points of the Table. However, as indicated by the values in column 8 of the Table, the figure of merit values for some data points are relatively low. Whereas this may be taken into account in the processing of the data for the production of an electron density map, an alternative would be to ignore one or more of the data points associated with low merit values. Thus it will be understood by those of skill in the art that reference to the data of Table 3 includes the situation where a small fraction (less than 5% and preferably less than 1%, such as less than 0.5%) of the data point rows are not utilised.

Once interpretation of the current map has been completed to provide an electron density map it is possible to combine the experimental phase with phases derived from the model and thus generate a new electron density map that will allow most of the crystal structure to be defined.

From the electron density map provided herein one can obtain the co-ordinate data of the 3A4 crystal structure. An electron density map is interpreted by placing an atomic structure in the model such that the model fits the map. An assessment of how the model agrees with the map can be derived by calculating a correlation coefficient between the map and the transformed model, calculation of a 2Fo-Fc map or generation of Rfactor and Free R factors by a refinement protocol. Partial interpretation of the electron density map at high resolutions (e.g. 2.5-1.0 Å) can be automated in the case of high quality maps. For a lower resolution map (e.g. less than 2.5Å), or maps generated from phases with less than ideal phasing statistics, interpretation is more subjective and may require manual input. The coordinates then obtained from this provide a measure of atomic location in Ångstroms. The coordinates are a relative set of positions that define a shape in three dimensions, but the skilled person would understand that an entirely different set of coordinates having a different origin and/or axes could define a similar or identical shape. Furthermore, the skilled person would understand that varying the relative atomic positions of the atoms of the structure so that the root mean square deviation of the residue backbone atoms (i.e. the nitrogen-carbon-carbon backbone atoms of the protein amino acid residues) is less than 2.0 Å, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å and most preferably less than 0.5 Å when superimposed on the coordinates derived from the data in Table 3 for the residue backbone atoms, will generally result in a structure which is substantially the same as the structure derived from of Table 3 in terms of both its structural characteristics and usefulness for structure-based analysis of P450-interactivity molecular structures.

Likewise the skilled person would understand that changing the number and/or positions of the water molecules and/or substrate molecules available from the electron density map from Table 3 will not generally affect the usefulness of the structure for structure-based analysis of P450-interacting structure. Thus for the purposes described herein as being aspects of the present invention, it is within the scope of the invention if: the coordinates available from Table 3 are transposed to a different origin and/or axes; the relative atomic positions of the atoms of the structure are varied so that the root mean square deviation of residue backbone atoms is less than 2.0 Å, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å when superimposed on the coordinates for the residue backbone atoms; and/or the number and/or positions of water molecules and/or substrate molecules is varied.

Reference herein to the coordinate data derived from Table 3 and the like thus includes the coordinate data in which one or more individual values of the Table are varied in this way. By "root mean square deviation" we mean the square root of the arithmetic mean of the squares of the deviations from the mean.

Those of skill in the art will appreciate that in many applications of the invention, it is not necessary to utilise all the coordinates of a structure derived using the data in Table 3, but merely a portion of them. Such a portion of coordinates is also referred to herein as "selected coordinates". For example, as described below, in methods of modelling candidate compounds with 3A4, selected coordinates of 3A4 may be used, for example at least 5, preferably at least 10, more preferably at least 50 and even more preferably at least 100 atoms such as at least 500 or at least 1,000 of the 3A4 structure. Likewise, the other applications of the invention described herein, including homology modelling and structure solution, and data storage and computer assisted manipulation of the coordinates, may also utilise all or a portion of the coordinates from the electron density map in Table 3 may be used.

Also, modifications in the 3A4 crystal structure due to e.g. mutations, additions, substitutions, and/or deletions of amino acid residues (including the deletion of one or more 3A4 protomers) could account for variations in the 3A4 atomic coordinates. However, atomic coordinate data of 3A4 modified so that a ligand that bound to one or more binding sites of 3A4 would be expected to bind to the corresponding binding sites of the modified 3A4 are, for the purposes described herein as being aspects of the present invention, also within the scope of the invention. Reference herein to the coordinates from the electron density map from Table 3 thus includes the coordinates modified in this way. Preferably, the modified data define at least one 3A4 binding cavity.

By providing structure factor phase data, the present invention allows electron density models of other 3A4 crystals or crystals of homologous proteins to be obtained without the need to perform multiple anomalous diffraction (MAD) structure determination or MIR. Thus the invention provides a method of determining an electron density map of a target protein which is, or is homologous to, 3A4, which method comprises providing a crystal of the target protein, obtaining an X-ray diffraction of said protein, and generating an electron density map of said target protein by reference to the structure factor phase data of Table 3.

Analysis of phase differences is applicable when the crystal forms are the same. A more general method of comparing structures is based on an analysis of electron density maps. Therefore preferably, for the purposes of this invention two maps are considered to be equivalent if the linear correlation coefficient calculated for the maps is greater than 0, and more preferably greater than 0.25 or 0.5. If the electron densities of two maps are respectively defined by the variates ρ_1 and ρ_2, the linear correlation coefficient, CC, is defined as:

$$CC(\rho_1, \rho_2) = \frac{\sum(\rho_1 - \bar{\rho}_1)(\rho_2 - \bar{\rho}_2)}{\sqrt{\sum(\rho_1 - \bar{\rho}_1)^2 \sum(\rho_2 - \bar{\rho}_2)^2}}$$

where $\bar{\rho}_1$ and $\bar{\rho}_2$ are the respective average densities of the two maps. Clearly, if two maps are identical, the CC takes a value of 1. More detail regarding the use of the CC as a quantifier of the similarity of electron density maps is provided in Section K below.

To compute the CC for two maps, the following procedure may be used. Firstly, for each map a molecular (i.e. P450 3A4) mask is determined. This can be done, for example, using the CCP4 DM program to distinguish the P450 3A4 molecule from the solvent region. Each grid point within a molecular boundary is labelled '1' and each grid point outside a boundary is labelled '0'. One map is then transformed into maximum coincidence with the other map. This is accomplished, for example, using the CCP4 FFFEAR program to search for a best fit between two maps. During the transformation, rigid-body translations and rotations are allowed. One of the maps and the corresponding mask are then interpolated onto the grid points of the other map and mask. The interpolation can be performed using the Astex-ROTMAP program provided in

Annex 1. Finally, the CC is computed for the masked maps, e.g. using the Astex-DENCOR program provided in Annex 2.

Furthermore, for the purposes of this invention an electron density map generated from the data of Table 3 and a set of atomic coordinates are considered to be equivalent if the CC calculated for the map generated from the data of Table 3 and a further electron density map generated from the atomic coordinate data is greater than 0, and more preferably greater than 0.25 or 0.5.

The computation of the CC in this case can follow the procedure discussed above with the additional prior step of generating the further electron density map from the atomic coordinate data. The generation can be conveniently performed using the standard CCP4 programs REFMAC and FFT to respectively calculate structure factors and then electron densities.

C. Crystal Coordinates.

In a further aspect, the invention also provides a crystal of P450 having the three dimensional atomic coordinates of Table 5. The atomic coordinates of Table 5 exclude residues from a loop region (261-270), which are not as clear and amenable for unambiguous interpretation as other regions of the protein. It is not unconceivable that this loop may adopt a different conformation under different conditions e.g. data from a different crystal, upon additional of compound, and the like. Crystals of the invention will thus comprise the coordinates of Table 5, with the coordinates of the loop region optionally being as further described herein, though other atomic coordinates for this loop region are not excluded.

An advantageous feature of the structure defined by the atomic coordinates of Table 5 is that it has a resolution of about 2.8 Å. More particularly, the residues in the binding pocket are well resolved.

A further advantage of the 3A4 structure described herein is that it is an unliganded, apo structure. This makes it particularly suitable for soaking in ligands and hence determining co-complex structures and is also ideal for homology modelling purposes as there is no conformational bias from a ligand.

Tables 5 and 6 gives atomic coordinate data for P450 3A4. In Tables 5 and 6 the third column denotes the atom, the fourth the residue type, the fifth the chain identification (in this case, chain A), the sixth the residue number (the atom numbering is with respect to the full length wild type protein), the seventh, eighth and ninth columns are the X, Y, Z coordinates respectively of the atom in question, the tenth column the occupancy of the atom, the eleventh the temperature factor of the atom, the twelfth the atom type.

Tables 5 and 6 are set out in an internally consistent format. For example (except in the case of Tyr 25), the coordinates of the atoms of each amino acid residue are listed such that the backbone nitrogen atom is first, followed by the C-alpha backbone carbon atom, designated CA, followed by side chain residues (designated according to one standard convention) and finally the carbon and oxygen of the protein backbone. Alternative file formats (e.g. such as a format consistent with that of the EBI Macromolecular Structure Database (Hinxton, UK)) which may include a different ordering of these atoms, or a different designation of the side-chain residues or haem molecule atoms, may be used or preferred by others of skill in the art. However it will be apparent that the use of a different file format to present or manipulate the coordinates of the Table is within the scope of the present invention.

The coordinates of Tables 5 and 6 provide a measure of atomic location in Angstroms, to 3 decimal places. The coordinates are a relative set of positions that define a shape in three dimensions, but the skilled person would understand that an entirely different set of coordinates having a different origin and/or axes could define a similar or identical shape. Furthermore, the skilled person would understand that varying the relative atomic positions of the atoms of the structure so that the root mean square deviation of the residue backbone atoms (i.e. the nitrogen-carbon-carbon backbone atoms of the protein amino acid residues) is less than 2.0 Å, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å when superimposed on the coordinates provided in Table 5 or 6 for the residue backbone atoms, will generally result in a structure which is substantially the same as the structure of Tables 5 or 6 in terms of both its structural characteristics and usefulness for structure-based analysis of P450-interactivity molecular structures.

Likewise the skilled person would understand that changing the number and/or positions of the water molecules molecules of Table 5 will not generally affect the usefulness of the structure for structure-based analysis of P450-interacting structure. Thus for the purposes described herein as being aspects of the present invention, it is within the scope of the invention if: the Tables 5 or 6 coordinates are transposed to a different origin and/or axes; the relative atomic positions of the atoms of the structure are varied so that the root mean square deviation of residue backbone atoms is less than 2.0 Å, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å when superimposed on the coordinates provided in Tables 5 or 6 for the residue backbone atoms; and/or the number and/or positions of water molecules is varied.

Reference herein to the coordinate data of Tables 5 or 6 and the like thus includes the coordinate data in which one or more individual values of the Table are varied in this way. By "root mean square deviation" we mean the square root of the arithmetic mean of the squares of the deviations from the mean.

With regard to the loop region referred to above, comparision of the different P450 structures determined to date indicates that various loops within the proteins can adopt very different conformations, often in response to compound binding. In the apo form of 3A4 which has been crystallized herein, a possible form of the loop region 261-270 is set out in Table 6. Thus in one aspect, the invention provides a crystal of P450 comprising amino acids having the atomic coordinates of Table 5, wherein the crystal additionally comprises amino acids having the atomic coordinates of Table 6.

Unless explicitly set out to the contrary, or otherwise clear from the context, reference throughout the present specification to the use of all or selected coordinates of or from Table 5 does not exclude the use of additional coordinates, particularly some or all of the coordinates of Table 6.

Protein structure similarity is routinely expressed and measured by the root mean square deviation (r.m.s.d.), which measures the difference in positioning in space between two sets of atoms. The r.m.s.d. measures distance between equivalent atoms after their optimal superposition. The r.m.s.d. can be calculated over all atoms, over residue backbone atoms (i.e. the nitrogen-carbon-carbon backbone atoms of the protein amino acid residues), main chain atoms only (i.e. the nitrogen-carbon-oxygen-carbon backbone atoms of the protein amino acid residues), side chain atoms only or more usually over C-alpha atoms only. For the purposes of this invention, the r.m.s.d. can be calculated over any of these, using any of the methods outlined below.

Methods of comparing protein structures are discussed in Methods of Enzymology, vol 115, pg 397-420. The necessary least-squares algebra to calculate r.m.s.d. has been given by Rossman and Argos (J. Biol. Chem. , vol 250, pp7525 (1975)) although faster methods have been described by Kabsch (Acta Crystallogr., Section A, A92, 922 (1976)); Acta Cryst. A34, 827-828 (1978)), Hendrickson (Acta Crystallogr., Section A, A35, 158 (1979)); McLachan (J. Mol. Biol., vol 128, pp49 (1979)) and Kearsley (Acta Crystallogr., Section A, A45, 208 (1989)). Some algorithms use an iterative procedure in which the one molecule is moved relative to the other, such as that described by Ferro and Hermans (Ferro and Hermans, Acta Crystallographic, A33, 345-347 (1977)). Other methods e.g. Kabsch's algorithm locate the best fit directly.

Programs for determining r.m.s.d include MNYFIT (part of a collection of programs called COMPOSER, Sutcliffe, M.J., Haneef, I., Carney, D. and Blundell, T.L. (1987) Protein Engineering, 1, 377-384), MAPS (Lu, G. An Approach for Multiple Alignment of Protein Structures (1998, in manuscript and on http://bioinfo1.mbfys.lu.se/TOP/maps.html)).

It is usual to consider C-alpha atoms and the rmsd can then be calculated using programs such as LSQKAB (Collaborative Computational Project 4. The CCP4 Suite: Programs for Protein

Crystallography, *Acta Crystallographica*, D50, (1994), 760-763), QUANTA (Jones et al., Acta Crystallography A47 (1991), 110-119 and commercially available from Accelerys, San Diego, CA), Insight (commercially available from Accelerys, San Diego, CA), Sybyl® (commercially available from Tripos, Inc., St Louis), O (Jones et al., *Acta Crystallographica*, A47, (1991), 110-119), and other coordinate fitting programs.

In, for example the programs LSQKAB and O, the user can define the residues in the two proteins that are to be paired for the purpose of the calculation. Alternatively, the pairing of residues can be determined by generating a sequence alignment of the two proteins, programs for sequence alignment are discussed in more detail in Section G. The atomic coordinates can then be superimposed according to this alignment and an r.m.s.d. value calculated. The program Sequoia (C.M. Bruns, I. Hubatsch, M. Ridderström, B. Mannervik, and J.A. Tainer (1999) Human Glutathione Transferase A4-4 Crystal Structures and Mutagenesis Reveal the Basis of High Catalytic Efficiency with Toxic Lipid Peroxidation Products, *Journal of Molecular Biology* 288(3): 427-439) performs the alignment of homologous protein sequences, and the superposition of homologous protein atomic coordinates. Alternatively, the program Astex-KFIT (see Annex 4) can be used. Once aligned, the r.m.s.d. can be calculated using programs detailed above. For sequence identical, or highly identical, the structural alignment of proteins can be done manually or automatically as outlined above. Another approach would be to generate a superposition of protein atomic coordinates without considering the sequence.

It is more normal when comparing significantly different sets of coordinates to calculate the r.m.s.d. value over C-alpha atoms only. It is particularly useful when analysing side chain movement to calculate the r.m.s.d. over all atoms and this can be done using LSQKAB and other programs.

Thus, for example, varying the atomic positions of the atoms of the structure by up to about 0.5 Å, preferably up to about 0.3 Å in any direction will result in a structure which is substantially the same as the structure of Table 5 in terms of both its structural characteristics and utility e.g. for molecular structure-based analysis.

Those of skill in the art will appreciate that in many applications of the invention, it is not necessary to utilise all the coordinates of Table 5, but merely a portion of them. For example, as described below, in methods of modelling candidate compounds with P450, selected coordinates of 3A4 may be used.

By "selected coordinates" it is meant for example at least 5, preferably at least 10, more preferably at least 50 and even more preferably at least 100, for example at least 500 or at least 1000 atoms of the 3A4 structure. Likewise, the other applications of the invention described herein, including homology modelling and structure solution, and data storage and computer assisted manipulation of the coordinates, may also utilise all or a portion of the coordinates (i.e.

selected coordinates) of Table 5. The selected coordinates may include or may consist of atoms found in the 3A4 P450 binding pocket, as described herein below, and particularly those of Tables 7 and more particularly those of Table 8.

D. Description of Structure.

In the structure of 3A4 set out herein, the first resolvable residue is Tyr25 and the last residue Gly498 (the protein as purified comprises residues residues 1, 2, and 25-503 of the wild type sequence (using wild type numbering from M18907) and a four histidine tag as shown in SEQ ID 2). The overall fold of the protein is typical of all P450 structures solved to date and the secondary structure elements are named according to the convention adopted for P450s Ravichandran, K. G., Boddupalli, S. S., Hasermann, C. A., Peterson, J. A., and Deisenhofer, J. (1993) *Science 261*, 731-736. The haem sits centrally within the molecule with the single cysteine 442 coordinating and hydrogen bonds between the haem propionates and Arg105, Trp126, Arg375 and Arg440.

There are a number of distinguishing differences between previously solved P450 structures and the structure of 3A4. There is a short helix towards the N terminus (here denoted helix A''), not observed previously the mammalian P450 structures, before helix A'. The B-C loop has less helical nature in 3A4 than in the previously solved human P450 2C9 structure (as contained in WO 03/035693 A2). This region along with the F-G loop region, has been implicated in forming an access channel (Podust, L. M., Stojan, J., Poulos, T. L., and Waterman, M. R. (2001) *J Inorg Biochem 87*, 227-235).

There are also some differences in the F helix (which is shorter than in the 2C9 structure), the F' helix and G' helix (which is shorter). The FG loop comprised 34 residues (210-243) and includes helix F' and helix G', compared to the 23 residues in the FG loop of 2C9. When compared to other P450s, the long FG loop of 3A4 is more due to the shortness of helix F than to the length of the FG loop itself. The B-C and F-G loops are in close proximity, forming two sides of the active site. It is widely accepted that 3A4 may bind several compound simultaneously, and can bind large compounds in excess of 1000 Da (e.g. erythromycin). Movement of these regions may be required to allow the compound entry and egress, and they may become more structured if in alternative conformations. The loops between helices G and H, and helices H and I are not clearly resolved in the electron density maps (residues 261-270, 277-290) and have been excluded from the model.

The dominating feature of the active site of substrate-free 3A4 is the cluster of phenylalanine residues (Phe57, Phe108, Phe213, Phe215, Phe219, Phe220, Phe241, Phe304) above the haem. Of these, some have been implicated by site directed mutagenesis to play a role in cooperativity and stereoselectivity. The majority of these residues lie within substrate region sites (SRS) (Gotoh, O. (1992) *J Biol Chem 267*, 83-90) first identified for the CYP 2C family of proteins.

Another cluster of four phenylalanine residues is found just below and to the side of the haem itself, in a position less clearly important for compound binding.

The kinetics exhibited by 3A4 can be complicated, with many literature examples citing one or more compound being accommodated simultaneously within the active site of 3A4 (Domanski et al, Biochemistry 2001, 40, 10150-10160). Site directed mutagenesis suggests that different substrates may bind at different regions of the active site. There is also evidence for homotropic cooperativity (interactions between a substrate and one or more effector molecules of the same chemical structure) and hetertropic cooperativity (where the substrate and effector molecules have different chemical structures).

Identification and use of P450 binding pocket residues.

The crystal structure for 3A4 has for the first time allowed the precise identification of all the residues that line the binding site of the enzyme (Table 7). Some residues proposed to be in the catalytic site by a variety of sources can now be shown not to be binding pocket residues but residues that hold the catalytic residues in place.

Table 7 below details all the residues that line the binding site of 3A4.

Phe 57	Asp 76	Val 81	Asn 104	Arg 105	Arg 106
Pro 107	Phe 108	Gly 109	Pro 110	Val 111	Met 114
Ser 116	Ala 117	Ile 118	Ser 119	Ile 120	Glu 122
Thr 207	Leu 210	Leu 211	Phe 215	Phe 220	Leu 221
Ile 223	Thr 224	Ile 230	Glu 234	Val 235	Leu 236
Ile 238	Cys 239	Phe 241	Pro 242	Ala 297	Ile 301
Phe 302	Ile 303	Phe 304	Ala 305	Gly 306	Glu 308
Thr 309	Ser 312	Val 313	Pro 368	Ile 369	Ala 370
Met 371	Arg 372	Leu 373	Glu 374	Arg 375	Ser 398
Gly 481	Leu 482	Leu 483	Glu 484		

Some of these residues have previously been inferred to be in the binding site of 3A4 from modelling (e.g. homology modelling, SRS proposals, 3D/4D-QSAR, sequence alignments, or mutagenesis studies) which with the aid of the crystal structure can now be known to line the 3A4 binding pocket. Some residues found in the binding pocket have never before been identified as binding site residues. These are listed in Table 8. The identification of these will greatly facilitate the modelling of compound binding.

Table 8: Residues newly identified as lining the 3A4 binding pocket

Phe 57	Asp 76	Val 81	Arg 106	Gly 109	Pro 110
Val 111	Ser 116	Ala 117	Ile 118	Glu 122	Thr 207
Phe 220	Leu 221	Ile 223	Thr 224	Ile 230	Glu 234
Val 235	Leu 236	Cys 239	Phe 241	Pro 242	Ala 297

Phe 302	Ile 303	Gly 306	Ser 312	Val 313	Pro 368
Arg 372	Ser 398	Gly 481	Leu 482	Leu 483	Glu 484

Accordingly, in a preferred aspect of the invention, where the invention contemplates the use of selected coordinates in a method of the invention, such selected coordinates will comprise at least one coordinate, preferably at least one side-chain coordinate of an amino acid residue selected from either Table 7 or 8.

Preferably, the selected coordinates include the coordinates of all the atoms of Table 5 or Table 6 relating to at least one amino acid from Table 7 or 8.

Also preferred, whether all or just some atoms of a particular amino acid are selected, is that at least 2, more preferably at least 5, and most preferably at least 10 of the selected coordinates are of side chain residues from the corresponding number of different amino acid residues. These may be selected exclusively from either of Table 7 or 8, or a combination thereof. Preferably at least one side chain residue coordinate of Table 8 is included.

E. Chimeras.

The use of chimeric proteins to achieve desired properties is now common in the scientific literature. For example, Sieber et al (Nature Biotechnology (2001) 19, 456-460) produced hybrids between human cytochrome P450 isoform 1A2 and the bacterial P450 BM3, in order to make proteins with the specificity of 1A2, but which had desirable expression and solubility properties of BM3. Active site chimeras are also described: for example, Swairjo et al (Biochemistry (1998) 37, 10928-10936) made loop chimeras of HIV-1 and HIV-2 protease to try to understand determinants of inhibitor-binding specificity.

Of particular relevance are cases where the active site is modified so as to provide a surrogate system to obtain structural information. Thus Ikuta et al (J Biol Chem (2001) 276, 27548-27554) modified the active site of cdk2, for which they could obtain structural data, to resemble that of cdk4, for which no X-ray structure is currently available. In this way they were able to obtain protein/ligand structures from the chimaeric protein which were useful in cdk4 inhibitor design. In a similar way, based on comparison of primary sequences of highly related isoforms (such as 3A1, 3A5, 3A7, 3A12 or 3A43) the active site of the 3A4 protein could be modified to resemble those isoforms. Protein structures or protein/ligand structures of the chimaeric proteins could be used in structure-based alteration of the metabolism of compounds which are substrates of that related P450 isoform.

Even if the percentage of the amino acid sequence identity between mammalian P450 ranks from 20 to 80%, the overall folding of mammalian P450s is expected to be very similar, with the same spatial distribution of the structural elements. Furthermore, this class of enzymes exhibits distinct substrate specificities that rely on only a limited number of residues located in non-contiguous parts of the polypeptide chain. The substrate-binding pocket of P450 is generally constituted by residues that fall in the SRS regions (substrate recognition sites) defined by Gotoh (Gotoh, O, J. Biol. Chem, 267; 83-90 (1992)) and in loops of the molecule.

(i) Converting other P450 Proteins to 3A4-like chimeras

Aspects of the present invention therefore relate to modification of P450 proteins such that the active sites mimic those of related isoforms. For example, from a knowledge of the structure and residues of the active site of the human 3A4 structure contained herein, a person skilled in the art could modify a P450 protein such that the active site mimicked that of human 3A4. This protein could then be used to obtain information on compound binding through the determination of protein/ligand complex structures using the chimaeric P450 protein.

For example, in one aspect the present invention provides a chimaeric protein having a binding cavity which provides a substrate specificity substantially identical to that of P450 3A4 protein, wherein the chimaeric protein binding cavity is lined by a plurality of atoms which correspond to selected P450 3A4 atoms lining the P450 3A4 binding cavity, and the relative positions of the plurality of atoms corresponding to the relative positions, as defined by Table 5, of the selected P450 3A4 atoms.

It is possible to postulate that only few changes would be required to inter-convert the substrate specificities of P450 isoforms that exhibit more than 70% of amino acid identity. 3A4 is 89% identical to 3A7, and 3A43 shares 76, 76, and 71% sequence identity on the amino acid level with CYP3A4, 3A5, and 3A7, respectively (Westlind et al, Biochemical and Biophysical Research Communications (2001), 281(5), 1349-1355; Gellner et al, Pharmacogenetics (2001), 11(2), 111-121). For example, although 3A4 and 3A5 are 84% identical they exhibit clear substrate specificity differences (Aoyama T; Yamano S; Waxman D J; Lapenson D P; Meyer U A; Fischer V; Tyndale R; Inaba T; Kalow W; Gelboin H V; Journal Of Biological Chemistry (1989 Jun 25), 264(18), 10388-95). CYP3A4 is inhibited by mifepristone and yet CYP3A5 is not. Using a panel of 3A4/3A5 chimaeric proteins, Khan et al (Khan, Kishore K.; He, You Qun; Correia, Maria Almira; Halpert, James R; Drug Metabolism and Disposition (2002), 30(9), 985-990) have identified the sequence differences that explain the lack of inhibition of CYP3A5. These studies have demonstrated the feasibility of the transfer of substrate specificities between 3A4 and 3A5 by mutating residues within the SRS regions. CYP3A4 and CYP3A5 also show different regioselectivity towards aflatoxin B1 (AFB1) biotransformation, and a site-directed mutagenesis program to understand the structural features responsible for these differences, concluded that residues within the SRS region 2 alone were responsible for these differences (Huifen Wang, Ryan Dick, Hequn Yin, Estefania Licad-Coles, Deanna L. Kroetz, Grazyna Szklarz, Greg Harlow, James R. Halpert, and Maria Almira Correia, Biochemistry, 37 (36), 12536 -12545, 1998).

The substrate specificity of an enzyme generally relies on only a limited number of residues located in non-contiguous parts of the polypeptide chain. The substrate specificities of these isoforms could be analysed by substituting these residues by site-directed mutagenesis. The minimal changes that would be required to convert another P450 protein into a 3A4-like chimera could be at least two amino acids selected from binding pocket, particularly the amino acid binding pocket residues of Table 7 or 8, more preferably Table 8. These mutations can be introduced by site-directed mutagenesis e.g. using a Stratagene QuikChange^{TM} Site-Directed Mutagenesis Kit or cassette mutagenesis methods (Ausubel, F.M., Brent, R., Kingston, R.E. et al. editors. Current Protocols in Molecular Biology. John Wiley & Sons, Inc., New York, Sambrook, J., Fritsch, E.F., and Maniatis, T. (1989). Molecular Cloning: a Laboratory Manual. 2nd ed. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY.). Thus the invention provides a chimaeric protein having one or more binding pockets defined by the residues of Table 5 and preferably including some or all of the binding pocket residues of Tables 7 or 8.

(ii) Converting 3A4 to other 3A isoforms

This strategy could clearly be applied for proteins that exhibit high sequence homology with or without overlapping substrate specificities and from different species. The use of the crystal structure solved for 3A4 would allow the characterization of the binding mode of a variety of molecules in the substrate pocket of these proteins. This in turn would allow the identification of residues to be modified in the human isoforms to convert them into metabolising enzymes with different substrate or regioselective preferences.

In one embodiment, a chimaeric 3A4 enzyme is produced which is isoformal with another enzyme of the 3A subfamily. For example, 3A4 could be turned into a 3A1-like, 3A5-like, 3A7-like, 3A12-like or 3A43-like isoform with a few amino acid changes. Based on the information available from the literature on the structure/activity studies performed on the human 3A4, 3A5, 3A7 and 3A43 isoforms, and the analysis of the structure of the human 3A4, we postulate that the 3A4 protein could be converted to a 3A5-like, 3A7-like or 3A43-like isoform with the substrate specificities attributed to 3A5, 3A7 or 3A43, 3A5 in particular based on the references above. The mutations can be introduced by site-directed mutagenesis or cassette mutagenesis methods, as described herein.

The crystallization of such chimeras and the determination of the three-dimensional structures relies on the ability of our 3A4 protein to yield crystals that diffract at high resolution. The aim is to modify the inside part of 3A4 to produce a new substrate binding site of 3A5, 3A7 or 3A43 without modifying the outside shell of the proteins that allow the protein to crystallize.

F. Homology Modelling.

The invention also provides a means for homology modelling of other proteins (referred to below as target P450 proteins). By "homology modelling", it is meant the prediction of related P450 structures based either on X-ray crystallographic data or computer-assisted *de novo* prediction of structure, based upon manipulation of the coordinate data derivable from the electron density map calculated from Table 3.

"Homology modelling" extends to target P450 proteins which are analogues or homologues of the 3A4 protein whose structure has been determined in the accompanying examples. It also extends to P450 protein mutants of 3A4 protein itself.

The term "homologous regions" describes amino acid residues in two sequences that are identical or have similar (e.g. aliphatic, aromatic, polar, negatively charged, or positively charged) side-chain chemical groups. Identical and similar residues in homologous regions are sometimes described as being respectively "invariant" and "conserved" by those skilled in the art.

In general, the method involves comparing the amino acid sequences of the 3A4 protein of SEQ ID 2 with a target P450 protein by aligning the amino acid sequences. Amino acids in the sequences are then compared and groups of amino acids that are homologous (conveniently referred to as "corresponding regions") are grouped together. This method detects conserved regions of the polypeptides and accounts for amino acid insertions or deletions.

Homology between amino acid sequences can be determined using commercially available algorithms. The programs *BLAST*, *gapped BLAST*, *BLASTN*, *PSI-BLAST* and *BLAST 2* (provided by the National Center for Biotechnology Information) are widely used in the art for this purpose, and can align homologous regions of two amino acid sequences. These may be used with default parameters to determine the degree of homology between the amino acid sequence of the SEQ ID 2 protein and other target P450 proteins which are to be modelled.

Analogues are defined as proteins with similar three-dimensional structures and/or functions with little evidence of a common ancestor at a sequence level.

Homologues are defined as proteins with evidence of a common ancestor, i.e. likely to be the result of evolutionary divergence and are divided into remote, medium and close sub-divisions based on the degree (usually expressed as a percentage) of sequence identity.

A homologue is defined here as a protein with at least 15% sequence identity or which has at least one functional domain, which is characteristic of 3A4. This includes polymorphic forms of 3A4.

There are two types of homologue: orthologues and paralogues. Orthologues are defined as homologous genes in different organisms, i.e. the genes share a common ancestor coincident with the speciation event that generated them. Paralogues are defined as homologous genes in the same organism derived from a gene/chromosome/genome duplication, i.e. the common ancestor of the genes occurred since the last speciation event.

The homlogues could also be polymorphic forms of 3A4 such as alleles or mutants as described in section (A).

Once the amino acid sequences of the polypeptides with known and unknown structures are aligned, the structures of the conserved amino acids in a computer representation of the polypeptide with known structure are transferred to the corresponding amino acids of the polypeptide whose structure is unknown. For example, a tyrosine in the amino acid sequence of known structure may be replaced by a phenylalanine, the corresponding homologous amino acid in the amino acid sequence of unknown structure.

The structures of amino acids located in non-conserved regions may be assigned manually by using standard peptide geometries or by molecular simulation techniques, such as molecular dynamics. The final step in the process is accomplished by refining the entire structure using molecular dynamics and/or energy minimization.

Homology modelling as such is a technique that is well known to those skilled in the art (see e.g. Greer, *Science*, Vol. 228, (1985), 1055, and Blundell *et al.*, *Eur. J. Biochem*, Vol. 172, (1988), 513). The techniques described in these references, as well as other homology modelling techniques, generally available in the art, may be used in performing the present invention.

Homology modelling may be performed on a three dimensional atomic coordinate model of 3A4 obtained using the present invention. A preferred model is that of Table 5. Thus a person of skill in the art will be able to obtain a representation of the three dimensional structure of a crystal of cytochrome P450 3A4 by a method which comprises providing the data of at least columns 1, 2, 3, 6 and 7 of Table 3 and constructing an electron density map of said data. This method is optionally performed by reference to the data of column 8 of said Table. Having obtained an electron density map, the person of skill in the art will be able to generate an initial model of 3A4 fitted to said map, which may then be refined by reference to the data of columns

4 and 5 of said Table. Refinement may also take place of other models generated from other 3A4 crystal structures.

The refined data may then be used in a method which comprises calculating the three-dimensional coordinates of one or more atoms of 3A4 in said crystal to provide a first three dimensional structure of 3A4. The positions of one or more atoms in said first structure may be varied to provide a second structure with three-dimensional coordinates having a r.m.s.d of less than 2.0 Å from said first structure, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å. This may be performed for a variety of reasons, for example in the light of other P450 models, or to manually fit regions of 3A4 structures which may need to be further optimised.

Thus the invention provides a method of homology modelling comprising the steps of:

(a) aligning a representation of an amino acid sequence of a target P450 protein of unknown three-dimensional structure with the amino acid sequence of the P450 of SEQ ID 2 to match homologous regions of the amino acid sequences;

(b) modelling the structure of the matched homologous regions of said target P450 of unknown structure on the corresponding regions of the P450 structure as obtained as described above and/or that of Table 5 or selected coordinates thereof; and

(c) determining a conformation (e.g. so that favourable interactions are formed within the target P450 of unknown structure and/or so that a low energy conformation is formed) for said target P450 of unknown structure which substantially preserves the structure of said matched homologous regions.

Preferably one or all of steps (a) to (c) are performed by computer modelling.

The co-ordinate data obtained from the Table 3, e.g. that of Table 5 or selected coordinates thereof, will be particularly advantageous for homology modelling of other human P450 proteins, in particular human P450s such as 2C9, 2C19, 2D6, 3A5, 3A7, 1A1, 1A2, 2E1 preferably 3A5, 3A7 and 3A43. These proteins may be the target P450 protein in the method of the invention described above.

The aspects of the invention described herein which utilise the P450 structure *in silico* may be equally applied to homologue models of P450 obtained by the above aspect of the invention, and this application forms a further aspect of the present invention. Thus having determined a conformation of a P450 by the method described above, such a conformation may be used in a computer-based method of rational drug design as described herein.

G. Structure Solution

The electron density map of the human 3A4 P450 or the atomic coordinate data of 3A4 can also be used to solve the crystal structure of other target P450 proteins including other crystal forms of 3A4, mutants, co-complexes of 3A4, where X-ray diffraction data or NMR spectroscopic data of these target P450 proteins has been generated and requires interpretation in order to provide a structure.

In the case of 3A4, this protein may crystallize in more than one crystal form. The data of Tables 3 or 5, or portions thereof, as provided by this invention, are particularly useful to solve the structure of those other crystal forms of 3A4. It may also be used to solve the structure of 3A4 mutants, 3A4 co-complexes, or of the crystalline form of any other protein with significant amino acid sequence homology to any functional domain of 3A4.

In the case of other target P450 proteins, particularly the human P450 proteins referred to in Section F above, the present invention allows the structures of such targets to be obtained more readily where raw X-ray diffraction data is generated.

Thus, where X-ray crystallographic or NMR spectroscopic data is provided for a target P450 of unknown three-dimensional structure, the electron density map of P450, derived from Table 3, or the atomic coordinate data derived from Table 5, may be used to interpret that data to provide a likely structure for the other P450 by techniques which are well known in the art, e.g. phasing in the case of X-ray crystallography and assisting peak assignments in NMR spectra.

One method that may be employed for these purposes is molecular replacement. In this method, the unknown crystal structure, whether it is another crystal form of 3A4, a 3A4 mutant, a 3A4 chimera or an 3A4 co-complex, or the crystal of a target P450 protein with amino acid sequence homology to any functional domain of 3A4, may be determined using the 3A4 structure coordinates derivable from Table 3 or the coordinates of Table 5 of this invention. Furthermore, the electron density map as defined in Table 3 can be used directly for this purpose. This method will provide an accurate structural form for the unknown crystal more quickly and efficiently than attempting to determine such information *ab initio*.

Examples of computer programs known in the art for performing molecular replacement are CNX (Brunger A.T.; Adams P.D.; Rice L.M., Current Opinion in Structural Biology, Volume 8, Issue 5, October 1998, Pages 606-611 (also commercially available from Accelrys San Diego, CA), MOLREP (A.Vagin, A.Teplyakov, MOLREP: an automated program for molecular replacement, J. Appl. Cryst. (1997) 30, 1022-1025, part of the CCP4 suite) or AMoRe (Navaza, J. (1994). AMoRe: an automated package for molecular replacement. Acta Cryst. A50, 157-163).

Thus, in a further aspect of the invention provides a method for determining the structure of a protein, which method comprises;

providing the coordinates obtained from the electron density map of Table 3,

positioning the coordinates in the crystal unit cell of said protein so as to provide a structure for said protein.

Preferably the coordinates are those of Table 5 or selected coordinates thereof, which may include coordinates of atoms of the amino acid residues set out in Table 7 and more preferably in Table 8.

In a further aspect of the invention provides a method for determining the structure of a protein, which method comprises;

providing the structure factor and phases of Table 3,

positioning of a search model in the crystal unit cell of said protein so as to provide a structure for said protein.

The invention may also be used to assign peaks of NMR spectra of such proteins, by manipulation of the data of Tables 3 or 5.

In a preferred aspect of this invention the co-ordinates are used to solve the structure of target 3A4 particularly homologues of 3A4 for example P450s such as 3A5, 3A7 and 3A43.

H. Further Uses of Structure Factor and Phase data

The data contained within Table 3 allows for the calculation of an electron density map using the solvent flattened phases (column 7) and the weighted structure factors (column 6). In addition, the data allows for the calculation of an electron density map using the solvent flattened phases (column 7), the Figure of Merit (column 8) and the observed structure factor amplitudes (column 4).

The phases provided in Table 3 can also be used to calculate a map with the Figure of Merit and a different structure factor amplitude from a same or related crystal form of 3A4, or a same or related crystal form of a homologous protein.

All of these maps can be used for the phased molecular replacement of other homologous proteins, as discussed above in Section G, specifically 3A4 homologues.

Aspects of the present invention therefore are, methods of using the phases of Table 3 (reciprocal space) for:

a) calculating a map together with the solvent flattened structure factor amplitude (Table 3), or

b) calculating a map together with the figure-of-merit and the measured structure factor amplitude (Table 3), or

c) calculating a map together with the figure-of-merit (Table 3) and structure factor amplitudes from the same or related crystal form of 3A4 or a same or related crystal form of a 3A4 homologue, and

d) use of any of these resulting electron densities (real space) from step a), b) or c) for molecular replacement.

In addition the map calculated from these structure factors and phases could be used in cross crystal form averaging between different crystals forms of CYP 3A4. If a different crystal form of 3A4 or a crystal form of a 3A4 homologue was obtained, the data of Table 3 can be used in cross crystal averaging, in reciprocal space, to improve the phases of either crystal form.

Complexes can be crystallized and analysed, and difference Fourier electron density maps can be calculated based on X-ray diffraction patterns of soaked or co-crystallized 3A4 and the structure factor and phases of Table 3. The difference Fourier electron density maps can then be analysed to determine whether and where a particular compound binds to 3A4 and/or changes the conformation of 3A4.

Thus it is possible to screen for ligand binding by the use of the differences between the structure factors of Table 3 and the structure factors derived from crystals into which a ligand has been introduced by soaking or co-crystallisation. The phases of Table 3 can then be used to generate a difference map.

A further aspect of the invention is therefore the use of the phases of Table 3 for calculating the difference Fourier map to identify whether a ligand has bound and its mode of binding:

a) calculating a difference Fourier map (together with the figure-of-merit) between the measured amplitudes (as presented in Table 3) and structure factor amplitudes from a ligand co-complex, or

b) calculating a difference Fourier map (together with the figure-of-merit) between any two sets of structure factor amplitudes for detecting ligands and/or heavy atoms, or

c) calculating an anomalous Fourier map (together with the figure-of-merit) for any structure factor amplitudes for detecting ligands and/or heavy atoms which have an anomalous scattering contribution.

I. Computer Systems.

In another aspect, the present invention provides systems, particularly a computer system, the systems containing either (a) electron density map derivable from Table 3 or co-ordinate data therefrom, said data defining the three-dimensional structure of P450 or at least selected coordinates thereof; (b) structure factor data (where a structure factor comprises the amplitude and phase of the diffracted wave) for 3A4, said structure factor data being the data of Table 3; (c) atomic coordinate data of a target P450 protein generated by homology modelling of the target based on the coordinate data derivable from Table 3; (d) atomic coordinate data of a target P450 protein generated by interpreting X-ray crystallographic data or NMR data by reference to the electron density map according to Table 3 or co-ordinate data therefrom; or (e) structure factor data derivable from the atomic coordinate data of (c) or (d).

In a preferred aspect, the atomic coordinate data are the data of Table 5, or selected coordinates thereof.

For example the computer system may comprise: (i) a computer-readable data storage medium comprising data storage material encoded with the computer-readable data; (ii) a working memory for storing instructions for processing said computer-readable data; and (iii) a central-processing unit coupled to said working memory and to said computer-readable data storage medium for processing said computer-readable data and thereby generating structures and/or performing rational drug design. The computer system may further comprise a display coupled to said central-processing unit for displaying said structures.

The invention also provides such systems containing atomic coordinate data of target P450 proteins wherein such data has been generated according to the methods of the invention described herein based on the starting data provided by Table 3. In one aspect, such data are those of Table 5 or selected coordinates thereof.

Such data is useful for a number of purposes, including the generation of structures to analyse the mechanisms of action of P450 proteins and/or to perform rational drug design of compounds, which interact with P450, such as compounds, which are metabolised by P450s.

In a further aspect, the present invention provides computer readable media with at least one of (a) electron density map derivable from Table 3 or co-ordinate data therefrom, recorded thereon, said data defining the three-dimensional structure of P450, or at least selected coordinates thereof; (b) structure factor data for P450 recorded thereon, the structure factor data of Table 3; (c) atomic coordinate data of a target P450 protein generated by homology modelling of the target based on the coordinate data derivable from Table 3; (d) atomic coordinate data of a target P450 protein generated by interpreting X-ray crystallographic data or NMR data by reference to the data of Table 3; or (e) structure factor data derivable from the atomic coordinate data of (c) or (d). The atomic coordinate data may be that of Table 5, or selected coordinates thereof.

In another aspect, the invention provides a computer-readable storage medium, comprising a data storage material encoded with computer readable data, wherein the data are defined by all or a portion (e.g. selected coordinates as defined herein) of the structure coordinates of P450 of Table 5, or a homologue of said P450, wherein said homologue comprises backbone atoms that have a root mean square deviation from the Cα or backbone atoms (nitrogen-carbon$_\alpha$-carbon) of Table 5 of less than 2 Å, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å.

As used herein, "computer readable media" refers to any medium or media, which can be read and accessed directly by a computer. Such media include, but are not limited to: magnetic storage media such as floppy discs, hard disc storage medium and magnetic tape; optical storage media such as optical discs or CD-ROM; electrical storage media such as RAM and ROM; and hybrids of these categories such as magnetic/optical storage media.

By providing such computer readable media, the atomic coordinate data derived from Table 3 can be routinely accessed to model P450s or selected coordinates thereof. For example, RASMOL (Sayle et al., *TIBS*, Vol. 20, (1995), 374) is a publicly available computer software package, which allows access and analysis of atomic coordinate data for structure determination and/or rational drug design.

As used herein, "a computer system" refers to the hardware means, software means and data storage means used to analyse the atomic coordinate data derived from Table 3 (e.g. that of Table 5 or selected coordinates thereof), as well as the electron density map of Table 3 of the present invention. The minimum hardware means of the computer-based systems of the present invention comprises a central processing unit (CPU), input means, output means and data storage means. Desirably a monitor is provided to visualize structure data. The data storage means may be RAM or means for accessing computer readable media of the invention. Examples of such systems are microcomputer workstations available from Silicon Graphics Incorporated and Sun Microsystems running Unix based, Windows NT or IBM OS/2 operating systems.

In another aspect, the invention provides a computer-readable storage medium, comprising a data storage material encoded with computer readable data, wherein the data are defined by all or a portion (e.g. selected coordinates as defined herein) of the structure coordinates of 3A4 obtainable from the data of Table 3 (such as that of Table 5 or selected coordinates thereof), or the electron density map of Table 3, or a homologue of 3A4, wherein said homologue comprises backbone atoms that have a root mean square deviation from the backbone atoms (nitrogen-carbon$_\alpha$-carbon) of co-ordinate data generated from Table 3 of not more than 2.0 Å, preferably less than 1.55 or 1.5 Å, more preferably less than 1.0 Å, and most preferably less than 0.5 Å.

The invention also provides a computer-readable data storage medium comprising a data storage material encoded with a first set of computer-readable data comprising Table 3, Table 5 or selected coordinates thereof; which, when combined with a second set of machine readable data comprising an X-ray diffraction pattern of a molecule or molecular complex of unknown structure, using a machine programmed with the instructions for using said first set of data and said second set of data, can determine at least a portion of the electron density corresponding to the second set of machine readable data.

A further aspect of the invention provides a method of providing data for generating structures and/or performing rational drug redesign with 3A4, 3A4 homologues or analogues, complexes of 3A4 with a compound, or complexes of 3A4 homologues or analogues with compounds, the method comprising:
(i) establishing communication with a remote device containing computer-readable data comprising at least one of: (a)) electron density map derivable from Table 3 or co-ordinate data therefrom, said data defining the three-dimensional structure of 3A4, at least one sub-domain of the three-dimensional structure of 3A4, or the coordinates of a plurality of atoms of 3A4; (b) structure factor data for 3A4, said structure factor data of Table 3; (c) atomic coordinate data of a target 3A4 homologue or analogue generated by homology modelling of the target based on the coordinate data derivable from Table 3; (d) atomic coordinate data of a protein generated by interpreting X-ray crystallographic data or NMR data by reference to the data of Table 3; and (e) structure factor data derivable from the atomic coordinate data of (c) or (d); and
(ii) receiving said computer-readable data from said remote device. The atomic coordinate data may be that of Table 5 or selected coordinates thereof.

Thus the remote device may comprise e.g. a computer system or computer readable media of one of the previous aspects of the invention. The device may be in a different country or jurisdiction from where the computer-readable data is received.

The communication may be via the internet, intranet, e-mail etc, transmitted through wires or by wireless means such as by terrestrial radio or by satellite. Typically the communication will be electronic in nature, but some or all of the communication pathway may be optical, for example, over optical fibers.

J. Uses of the Structures of the Invention.

The crystal structures obtained according to the present invention (including the structure derivable from Table 3 (e.g. that of Table 5 or selected coordinates thereof) as well as the structures of target P450 proteins obtained in accordance with the methods described herein), may be used in several ways for drug design. For example, many drugs or drug candidates fail to be of clinical use due to the detrimental interactions with P450 proteins, resulting in a rapid clearance of the drugs from the body. The present invention will allow those of skill in the art to attempt to rescue such compounds from development, by following the structure-based chemical strategies detailed below.

In the case where a drug molecule is being metabolised by a P450, information on the binding orientation by either co-crystallization, soaking or computationally docking the binding orientation of the drug in the binding pocket can be determined. This will guide specific modifications to the chemical structure designed to mediate or control the interaction of the drug with the protein. Such modifications can be designed with an aim to reduce the metabolism of the drug by P450 and so improve its therapeutic action.

The crystal structure could also be useful to understand drug-drug interactions. Many examples exist where adverse reactions to drugs are recorded if administered while the patient is already taking other medicines. The mechanism behind this detrimental and often dangerous drug-drug interaction scenario may be when one drug behaves as an inhibitor of a P450 resulting in toxic levels of the other drug building-up due to less or no metabolism occurring. The crystal structure of the present invention complexed to such an inhibitor (either *in* vitro or *in silico*) may also allow rational modifications either to modify the inhibitor such that it no longer inhibits or inhibits less, or to modify the second drug such that it could bind better to the P450 (so becoming metabolised) and so displace the inhibitor.

P450s display significant polymorphic variations dependent on the age, gender, or ethnic origin of the patient. This can manifest itself in adverse reactions from some segments of patient populations to some drugs. By using the crystal structures of the present invention to map the relevant mutation with respect to the binding mode of the drug, chemical modifications could also be made to the drug to avoid interactions with the variable region of the protein. This could ensure more consistent therapeutic value from the drug for such segments of the population and avoid dangerous side effects.

Some pharmaceutical compounds are converted by P450s into active metabolites. In the case of such compounds, a greater understanding of how such compounds are converted by a P450 will allow modification of the compound so that it can be converted at a different rate. For example, increasing the rate of conversion may allow a more rapid delivery of a desired therapeutic effect, whereas decreasing the rate of conversion may allow for higher doses to be administered or the development of sustained release pharmaceutical preparations, for example comprising a mixture of compounds which are metabolized at different rates to form the same active metabolite.

Thus, the determination of the three-dimensional structure of P450 provides a basis for the design of new compounds, which interact with P450 in novel ways. For example, knowing the three-dimensional structure of P450, computer modelling programs may be used to design different molecules expected to interact with possible or confirmed active sites, such as binding sites or other structural or functional features of P450.

(i) Obtaining and analysing crystal complexes.

In one approach, the structure of a compound bound to a P450 may be determined by experiment. This will provide a starting point in the analysis of the compound bound to P450, thus providing those of skill in the art with a detailed insight as to how that particular compound interacts with P450 and the mechanism by which it is metabolised.

Many of the techniques and approaches to structure-based drug design described above rely at some stage on X-ray analysis to identify the binding position of a ligand in a ligand-protein complex. A common way of doing this is to perform X-ray crystallography on the complex, produce a difference Fourier electron density map, and associate a particular pattern of electron density with the ligand. However, in order to produce the map (as explained e.g. by Blundell et al., in *Protein Crystallography*, Academic Press, New York, London and San Francisco, (1976)), it is necessary to know beforehand the protein 3D structure (or at least the protein structure factors). Therefore, determination of the P450 structure also allows difference Fourier electron density maps of P450-compound complexes to be produced, determination of the binding position of the drug and hence may greatly assist the process of rational drug design.

Accordingly, the invention provides a method for determining the structure of a compound bound to P450, said method comprising:

providing a crystal of P450 according to the invention;

soaking the crystal with said compounds; and

determining the structure of said P450 compound complex by employing the coordinate data derivable from Table 3 (e.g. that of Table 5 or selected coordinates thereof), or by employing the phases of Table 3, or by employing the electron density derivable from Table 3.

Alternatively, the P450 and compound may be co-crystallized. Thus the invention provides a method for determining the structure of a compound bound to P450, said method comprising; mixing the protein with the compound(s), crystallizing the protein-compound(s) complex; and determining the structure of said P450-compound(s) complex by reference to the coordinate data derivable from Table 3 (e.g. that of Table 5 or selected coordinates thereof), or by reference to the phases of Table 3, or by reference to the electron density derivable from Table 3.

The analysis of such structures may employ (i) X-ray crystallographic diffraction data from the complex and (ii) a three-dimensional structure of P450, or at least selected coordinates thereof, to generate a difference Fourier electron density map of the complex, the three-dimensional structure being defined by atomic coordinate data derivable from Table 3 (e.g. that of Table 5 or selected coordinates thereof), or by employing the phases of Table 3, or by employing the electron density derivable from Table 3. The difference Fourier electron density map may then be analysed.

Therefore, such complexes can be crystallized and analysed using X-ray diffraction methods, e.g. according to the approach described by Greer et al., *J. of Medicinal Chemistry*, Vol. 37, (1994), 1035-1054, and difference Fourier electron density maps can be calculated based on X-ray diffraction patterns of soaked or co-crystallized P450 and the solved structure of uncomplexed P450. These maps can then be analysed e.g. to determine whether and where a particular compound binds to P450 and/or changes the conformation of P450.

Electron density maps can be calculated using programs such as those from the CCP4 computing package (Collaborative Computational Project 4. The CCP4 Suite: Programs for Protein Crystallography, *Acta Crystallographica*, D50, (1994), 760-763.). For map visualization and model building programs such as "O" (Jones et al., *Acta Crystallographica*, A47, (1991), 110-119) can be used.

In addition, in accordance with this invention, 3A4 mutants may be crystallized in co-complex with known 3A4 substrates or inhibitors or novel compounds. The crystal structures of a series of such complexes may then be solved by molecular replacement and compared with that of the 3A4 structure from Table 3 or Table 5 or selected coordinates thereof. Potential sites for modification within the various binding sites of the enzyme may thus be identified. This information provides an additional tool for determining the most efficient binding interactions, for example, increased hydrophobic interactions, between 3A4 and a chemical entity or compound.

For example there are alleles of 3A4, which differ from the native 3A4 by only 1-2 amino acid substitutions, and yet individuals who express these allelic variants may exhibit very different drug metabolism profiles. Polymorphisms in the human CYP3A4 genes can influence the outcome of a treatment for a range of diseases including cancer. The metabolism of chemotherapeutic agents used in the treatment of cancer can be investigated using the structure provided here and the agents then altered using the methods described herein.

By generating such allelic proteins and determining the co-complex with compounds a greater understanding of allelic interactions with compounds may be developed.

All of the complexes referred to above may be studied using well-known X-ray diffraction techniques and may be refined against 1.5 to 3.5 Å resolution X-ray data to an R value of about 0.30 or less using computer software, such as CNX (Brunger et al., *Current Opinion in Structural Biology*, Vol. 8, Issue 5, October 1998, 606-611, and commercially available from Accelrys, San Diego, CA), and as described by Blundell et al, (1976) and Methods in Enzymology, vol. 114 & 115, H. W. Wyckoff et al., eds., Academic Press (1985).

This information may thus be used to optimise known classes of 3A4 substrates or inhibitors, and more importantly, to design and synthesize novel classes of 3A4 inhibitors and design drug with modified P450 metabolism.

(ii) In silico analysis and design

Although the invention will facilitate the determination of actual crystal structures comprising a P450 and a compound, which interacts with the P450, current computational techniques provide a powerful alternative to the need to generate such crystals and generate and analyse diffraction date. Accordingly, a particularly preferred aspect of the invention relates to *in silico* methods directed to the analysis and development of compounds which interact with P450 structures of the present invention.

Determination of the three-dimensional structure of 3A4 provides important information about the binding sites of 3A4, particularly when comparisons are made with similar enzymes. This information may then be used for rational design and modification of 3A4 substrates and inhibitors, e.g. by computational techniques which identify possible binding ligands for the binding sites, by enabling linked-fragment approaches to drug design, and by enabling the identification and location of bound ligands using X-ray crystallographic analysis. These techniques are discussed in more detail below.

Thus as a result of the determination of the P450 three-dimensional structure, more purely computational techniques for rational drug design may also be used to design structures whose interaction with P450 is better understood (for an overview of these techniques see e.g. Walters et al (*Drug Discovery Today*, Vol.3, No.4, (1998), 160-178; Abagyan, R.; Totrov, M. *Curr. Opin. Chem. Biol.* **2001**, *5*, 375-382). For example, automated ligand-receptor docking programs (discussed e.g. by Jones et al. in *Current Opinion in Biotechnology*, Vol.6, (1995), 652-656 and Halperin, I.; Ma, B.; Wolfson, H.; Nussinov, R. *Proteins* **2002**, *47*, 409-443), which require accurate information on the atomic coordinates of target receptors may be used.

The aspects of the invention described herein which utilize the P450 structure *in silico* may be equally applied to both the 3A4 structure from the data of Table 3 (e.g. that of Table 5 or selected coordinates thereof) and the models of target P450 proteins obtained by other aspects of the invention. Thus having determined a conformation of a P450 by the method described above, such a conformation may be used in a computer-based method of rational drug design as described herein. In addition the availability of the structure of the P450 3A4 will allow the generation of highly predictive pharmacophore models for virtual library screening or compound design.

Accordingly, the invention provides a computer-based method for the analysis of the interaction of a molecular structure with a P450 structure of the invention, which comprises:

providing the structure of a P450 of the invention;
providing a molecular structure to be fitted to said P450 structure; and
fitting the molecular structure to the P450 structure.

The P450 structure of the invention may be that of Table 5, or selected coordinates thereof.

In an alternative aspect, the method of the invention may utilize the coordinates of atoms of interest of the P450 binding region, which are in the vicinity of a putative molecular structure, for example within 10-25 Å of the catalytic regions or within 5-10 Å of a compound bound, in order to model the pocket in which the structure binds. These coordinates may be used to define a space, which is then analysed "*in silico*". Thus the invention provides a computer-based method for the analysis of molecular structures which comprises:

providing the coordinates of at least two atoms of a P450 structure of the invention ("selected coordinates");
providing the structure of a molecular structure to be fitted to said coordinates; and
fitting the structure to the selected coordinates of the P450.

In practice, it will be desirable to model a sufficient number of atoms of the P450 as defined by the coordinates derivable from Table 3 (e.g. those of Table 5 or selected coordinates thereof), which represent a binding pocket, e.g. the atoms of the residues identified in Tables 7 and 8, preferably Table 8. Binding pockets and other features of the interaction of P450 with co-factor are described in the accompanying example. Thus, in this embodiment of the invention, there will preferably be provided the coordinates of at least 5, preferably at least 10, more preferably at least 50 and even more preferably at least 100, e.g. at least 500 such as at least 1000, selected atoms of the P450 structure.

Although every different compound metabolised by P450 may interact with different parts of the binding pocket of the protein, the structure of this P450 allows the identification of a number of particular sites which are likely to be involved in many of the interactions of P450 with a drug candidate. The residues are set out in Tables 7 and 8. Thus in this aspect of the invention, the selected coordinates may comprise coordinates of some or all of these residues.

In order to provide a three-dimensional structure of compounds to be fitted to a P450 structure of the invention, the compound structure may be modelled in three dimensions using commercially available software for this purpose or, if its crystal structure is available, the coordinates of the structure may be used to provide a representation of the compound for fitting to a P450 structure of the invention.

The binding pockets of cytochrome P450 molecules are of a size which can accommodate more than one ligand. Indeed, some drug-drug interactions may occur as a result of interaction of the compounds within the binding pocket of the same P450. In any event, the findings of the present invention may be used to examine or predict the interaction of two or more separate molecular structures within the P450 3A4 binding pocket of the invention.

Thus the invention provides a computer-based method for the analysis of the interaction of two molecular structures within a P450 binding pocket structure, which comprises:

providing the P450 structure of Table 5 or selected coordinates thereof;
providing a first molecular structure;
fitting the first molecular structure to said P450 structure;
providing a second molecular structure; and
fitting the second molecular structure to a different part said P450 structure.

Optionally the method of analysis further comprises providing a third molecular structure and also fitting that structure to the P450 structure. Indeed, further molecular structures may be provided and fitted in the same way.

In one aspect, one or more of the molecular structures may be fitted to one or more of the phenylalanine residues of the 3A4 binding pocket mentioned above, and one or more of the other molecular structures may be fitted to coordinates of amino acids from another part of the P450 binding pocket, such as another part of the ligand-binding region, to the haem-binding region, or to atoms of the amino acid residues of Tables 7 or 8. In one embodiment, the one or more other molecular structures may be fitted, in addition to or instead of, to the haem structure in the P450 binding pocket.

Following the fitting of the molecular structures, a person of skill in the art may seek to use molecular modelling to determine to what extent the structures interact with each other (e.g. by hydrogen bonding, other non-covalent interactions, or by reaction to provide a covalent bond between parts of the structures) or the interaction of one structure with 3A4 is altered by the presence of another structure.

The person of skill in the art may use *in silico* modelling methods to alter one or more of the structures in order to design new structures which interact in different ways with 3A4, so as to speed up or slow down their metabolism, as the case may be.

Newly designed structures may be synthesised and their interaction with 3A4 may be determined or predicted as to how the newly designed structure is metabolised by said P450 structure. This process may be iterated so as to further alter the interaction between it and the 3A4.

By "fitting", it is meant determining by automatic, or semi-automatic means, interactions between at least one atom of a molecular structure and at least one atom of a P450 structure of the invention, and calculating the extent to which such an interaction is stable. Interactions include attraction and repulsion, brought about by charge, steric considerations and the like. Various computer-based methods for fitting are described further herein.

More specifically, the interaction of a compound or compounds with P450 can be examined through the use of computer modelling using a docking program such as GOLD (Jones et al., *J. Mol. Biol.*, 245, 43-53 (1995), Jones et al., *J. Mol. Biol.,* 267, 727-748 (1997)), GRAMM (Vakser, I.A., *Proteins* , Suppl., 1:226-230 (1997)), DOCK (Kuntz et al, *J.Mol.Biol.* **1982** , *161*, 269-288, Makino et al, *J.Comput.Chem.* **1997**, *18*, 1812-1825), AUTODOCK (Goodsell et al, *Proteins* **1990**, *8*, 195-202, Morris et al, *J.Comput.Chem.* **1998**, *19*, 1639-1662.), FlexX, (Rarey et al, *J.Mol.Biol.* **1996**, *261*, 470-489) or ICM (Abagyan et al, *J.Comput.Chem.* **1994**, *15*, 488-506). This procedure can include computer fitting of compounds to P450 to ascertain how well the shape and the chemical structure of the compound will bind to the P450.

Also computer-assisted, manual examination of the active site structure of P450 may be performed. The use of programs such as GRID (Goodford, *J. Med. Chem.*, 28, (1985), 849-857) - a program that determines probable interaction sites between molecules with various functional groups and an enzyme surface - may also be used to analyse the active site to predict, for example, the types of modifications which will alter the rate of metabolism of a compound.

Computer programs can be employed to estimate the attraction, repulsion, and steric hindrance of the two binding partners (i.e. the P450 and a compound).

If more than one P450 active site is characterized and a plurality of respective smaller compounds are designed or selected, a compound may be formed by linking the respective small compounds into a larger compound, which maintains the relative positions and orientations of the respective compounds at the active sites. The larger compound may be formed as a real molecule or by computer modelling.

Detailed structural information can then be obtained about the binding of the compound to P450, and in the light of this information adjustments can be made to the structure or functionality of the compound, e.g. to alter its interaction with P450. The above steps may be repeated and re-repeated as necessary.

As indicated above, molecular structures, which may be fitted to the P450 structure of the invention, include compounds under development as potential pharmaceutical agents. The agents may be fitted in order to determine how the action of P450 modifies the agent and to provide a basis for modelling candidate agents, which are metabolised at a different rate by a P450.

Molecular structures, which may be used in the present invention, will usually be compounds under development for pharmaceutical use. Generally such compounds will be organic molecules, which are typically from about 100 to 2000 Da, more preferably from about 100 to 1000 Da in molecular weight. Such compounds include peptides and derivatives thereof, steroids, anti-inflammatory drugs, anti-cancer agents, anti-bacterial or antiviral agents, neurological agents and the like. In principle, any compound under development in the field of pharmacy can be used in the present invention in order to facilitate its development or to allow further rational drug design to improve its properties.

(iii) Analysis and modification of compounds and metabolites

Where the primary metabolite of a potential or actual pharmaceutical compound is known, and this metabolite is generated by the action of P450, the structure of the agent and its metabolite may both be modelled and compared to each other in order to better determine residues of P450 which interact with the agent. In any event, the present invention provides a process for predicting potential pharmaceutical compounds with a desired activity which are metabolised by P450 at a rate different from a starting compound having the same desired activity, which method comprises:

fitting a starting compound to a P450 structure of the invention or selected coordinates thereof;

determining or predicting how said compound is metabolized by said P450 structure; and modifying the compound structure so as to alter the interaction between it and the P450.

It would be understood by those of skill in the art that modification of the structure will usually occur *in silico*, allowing predictions to be made as to how the modified structure interacts with the P450.

Modification will be those conventional in the art known to the skilled medicinal chemist, and will include, for example, substitutions or removal of groups containing residues which interact with the amino acid side chain groups of a P450 structure of the invention. For example, the replacements may include the addition or removal of groups in order to decrease or increase the charge of a group in a test compound, the replacement of a charge group with a group of the opposite charge, or the replacement of a hydrophobic group with a hydrophilic group or vice versa. It will be understood that these are only examples of the type of substitutions considered by medicinal chemists in the development of new pharmaceutical compounds and other modifications may be made, depending upon the nature of the starting compound and its activity.

Although it is usually desired to alter a compound to prevent its metabolism by P450, or at least to reduce the rate at which P450 metabolises the compound, the present invention also includes developing compounds which are metabolised more rapidly than a starting compound, for example where such a compound blocks metabolism of another drug.

Where a potential modified compound has been developed by fitting a starting compound to the P450 structure of the invention and predicting from this a modified compound with an altered rate of metabolism, the invention further includes the step of synthesizing the modified compound and testing it in a in vivo or in vitro biological system in order to determine its activity and/or the rate at which it is metabolised.

The above-described processes of the invention may be iterated in that the modified compound may itself be the basis for further compound design. The above-described processes may also be used to modify a compound which interacts with a second compound within the 3A4 binding pocket.

(iv) Analysis of compounds in binding pocket regions

Our finding of a cluster of phenylalanine residues in the vicinity of the haem of 3A4 allows the analysis and design methods described in the preceding subsections to be focused on compounds which interact with one or more of these residues.

For example, compounds which dock in the 3A4 substrate binding pocket in a manner which includes pi:pi stacking interactions with a phenylalanine side chain, may be modified in order to alter their metabolism. For example, such interactions may be influential in determining the rate at which the compounds undergo metabolism via movement towards, and reaction with, the haem moiety, located in the haem binding region of the 3A4 binding pocket. By altering (i.e. increasing or decreasing) their affinity of the compound to these phenylalanine residues, or other features of the ligand binding region compared to the haem binding region it may alter (i.e. increase or decrease) their ability to move towards, or be retained by, the haem-binding region.

For example by increasing their affinity to the ligand-binding region over the haem binding region may decrease their ability to move towards the haem-binding region. Alternatively, decreasing their affinity to the ligand-binding region may be desired to decrease their affinity to this region compared to the haem binding region and hence increase their ability to move towards the haem binding region. If compound binding to the ligand-binding pocket is a necessary prerequisite of compound binding in the haem-binding region and its subsequent metabolism by or inhibition of 3A4, elimination of binding to the ligand-binding region may eliminate all compound metabolism by 3A4 or inhibition of 3A4. An alternative or additional approach is to modify such substrates to increase or decrease their affinity for residues of the haem-binding region. Changes of this type may be introduced in order to increase or decrease the turnover of the substrates.

Some molecules are known to be effectors or activators of 3A4 metabolism. Modification of the binding between 3A4 and such a compound would mediate metabolism of the substrate.

Thus in one embodiment, the present invention provides a method for modifying the structure of a compound in order to alter its metabolism by a P450, which method comprises:

fitting a starting compound to one or more coordinates of at least one amino acid residue of the ligand-binding region of the P450;

modifying the starting compound structure so as to increase or decrease its interaction with the ligand-binding region;

wherein said ligand-binding region is defined as including at least one of the P450 residues numbered as Phe57, Phe108, Phe213, Phe215, Phe219, Phe220, Phe241 and Phe304.

In another embodiment, the present invention provides a method for modifying the structure of a compound in order to alter its metabolism by a P450, which method comprises:

fitting a starting compound to one or more coordinates of at least one amino acid residue of the ligand-binding region of the P450;

modifying the starting compound structure so as to increase or decrease its interaction with the ligand-binding region;

wherein said ligand-binding region is defined as including at least one of the P450 residues of Table 7 and preferably of Table 8.

In another embodiment, the invention provides a method for modifying the structure of a compound in order to alter its metabolism by a P450 3A4, which method comprises:

fitting a starting compound to one or more coordinates of at least one amino acid residue of the haem-binding region of the P450;

modifying the starting compound structure so as to increase or decrease its interaction with the haem-binding region.

The haem binding region also optionally includes the iron ion bound to the haem molecule, and if desired, one or more of the other atoms of the haem molecule itself. In a preferred aspect of the invention, the iron ion is also included in the haem-binding region.

Desirably, in the above aspects of the invention, coordinates from at least two, preferably at least five, and more preferably at least ten amino acid residues of the P450 (including where desired the iron ion) will be used.

For the avoidance of doubt, the term "modifying" is used as defined in the preceding subsection, and once such a compound has been developed it may be synthesised and tested also as described above.

(v) Fragment linking and growing.

The provision of the crystal structures of the invention will also allow the development of compounds which interact with the binding pocket regions of P450s (for example to act as inhibitors of a P450) based on a fragment linking or fragment growing approach.

For example, the binding of one or more molecular fragments can be determined in the protein binding pocket by X-ray crystallography. Molecular fragments are typically compounds with a molecular weight between 100 and 200 Da (Carr et al, 2002). This can then provide a starting point for medicinal chemistry to optimise the interactions using a structure-based approach. The fragments can be combined onto a template or used as the starting point for 'growing out' an inhibitor into other pockets of the protein (Blundell et al, 2002). The fragments can be positioned in the binding pocket of the P450 and then 'grown' to fill the space available, exploring the electrostatic, van der Waals or hydrogen-bonding interactions that are involved in molecular recognition. The potency of the original weakly binding fragment thus can be rapidly improved using iterative structure-based chemical synthesis.

At one or more stages in the fragment growing approach, the compound may be synthesized and tested in a biological system for its activity. This can be used to guide the further growing out of the fragment.

Where two fragment-binding regions are identified, a linked fragment approach may be based upon attempting to link the two fragments directly, or growing one or both fragments in the manner described above in order to obtain a larger, linked structure, which may have the desired properties.

Where the binding site of two or more ligands are determined they may be connected to form a potential lead compound that can be further refined using e.g. the iterative technique of *Greer* et al. For a virtual linked-fragment approach see Verlinde et al., *J. of Computer-Aided Molecular Design*, 6, (1992), 131-147, and for NMR and X-ray approaches see Shuker et al., *Science*, 274, (1996), 1531-1534 and Stout et al., *Structure*, 6, (1998), 839-848. The use of these approaches to design P450 inhibitors is made possible by the determination of the P450 structure.

(vi) Compounds of the invention.

Where a potential modified compound has been developed by fitting a starting compound to the P450 structure of the invention and predicting from this a modified compound with an altered rate of metabolism (including a slower, faster or zero rate), the invention further includes the step of synthesizing the modified compound and testing it in an in vivo or in vitro biological system in order to determine its activity and/or the rate at which it is metabolised.

The method comprises: (a) providing 3A4 under conditions where, in the absence of modulator, the 3A4 is able to metabolise known substrates; (b) providing the compound; and (c) determining the extent to which the compound is metabolised in the presence of 3A4 or (d) determining the extent to which the compound inhibits metabolism of a known substrate of 3A4.

More preferably, in the latter steps the compound is contacted with P450 under conditions to determine its function.

For example, in the contacting step above the compound is contacted with P450 in the presence of the compound, and typically a buffer and substrate, to determine the ability of said compound to inhibit P450 or to be metabolised by P450. The substrate may be e.g. dibenzylfluorescein. So, for example, an assay mixture for P450 may be produced which comprises the compound, substrate and buffer.

In another aspect, the invention includes a compound, which is identified by the methods of the invention described above.

Following identification of such a compound, it may be manufactured and/or used in the preparation, i.e. manufacture or formulation, of a composition such as a medicament, pharmaceutical composition or drug. These may be administered to individuals.

Thus, the present invention extends in various aspects not only to a compound as provided by the invention, but also a pharmaceutical composition, medicament, drug or other composition comprising such a compound. The compositions may be used. for treatment (which may include preventative treatment) of disease such as cancer. Such a treatment may comprise administration of such a composition to a patient, e.g. for treatment of disease; the use of such an inhibitor in the manufacture of a composition for administration, e.g. for treatment of disease; and a method of making a pharmaceutical composition comprising admixing such an inhibitor with a pharmaceutically acceptable excipient, vehicle or carrier, and optionally other ingredients.

Thus a further aspect of the present invention provides a method for preparing a medicament, pharmaceutical composition or drug, the method comprising:
(a) identifying or modifying a compound by a method of any one of the other aspects of the invention disclosed herein; (b) optimising the structure of the molecule; and (c) preparing a medicament, pharmaceutical composition or drug containing the optimised compound.

The above-described processes of the invention may be iterated in that the modified compound may itself be the basis for further compound design.

By "optimising the structure" we mean e.g. adding molecular scaffolding, adding or varying functional groups, or connecting the molecule with other molecules (e.g. using a fragment linking approach) such that the chemical structure of the modulator molecule is changed while its original modulating functionality is maintained or enhanced. Such optimisation is regularly undertaken during drug development programmes to e.g. enhance potency, promote pharmacological acceptability, increase chemical stability etc. of lead compounds.

Modification will be those conventional in the art known to the skilled medicinal chemist, and will include, for example, substitutions or removal of groups containing residues which interact with the amino acid side chain groups of a P450 structure of the invention. For example, the replacements may include the addition or removal of groups in order to decrease or increase the charge of a group in a test compound, the replacement of a charge group with a group of the opposite charge, or the replacement of a hydrophobic group with a hydrophilic group or vice versa. It will be understood that these are only examples of the type of substitutions considered by medicinal chemists in the development of new pharmaceutical compounds and other modifications may be made, depending upon the nature of the starting compound and its activity.

Compositions may be formulated for any suitable route and means of administration. Pharmaceutically acceptable carriers or diluents include those used in formulations suitable for oral, rectal, nasal, topical (including buccal and sublingual), vaginal or parenteral (including subcutaneous, intramuscular, intravenous, intradermal, intrathecal and epidural) administration. The formulations may conveniently be presented in unit dosage form and may be prepared by any of the methods well known in the art of pharmacy.

For solid compositions, conventional non-toxic solid carriers include, for example, pharmaceutical grades of mannitol, lactose, cellulose, cellulose derivatives, starch, magnesium stearate, sodium saccharin, talcum, glucose, sucrose, magnesium carbonate, and the like may be used. Liquid pharmaceutically administrable compositions can, for example, be prepared by dissolving, dispersing, etc, an active compound as defined above and optional pharmaceutical adjuvants in a carrier, such as, for example, water, saline aqueous dextrose, glycerol, ethanol, and the like, to thereby form a solution or suspension. If desired, the pharmaceutical composition to be administered may also contain minor amounts of non-toxic auxiliary substances such as wetting or emulsifying agents, pH buffering agents and the like, for example, sodium acetate, sorbitan monolaurate, triethanolamine sodium acetate, sorbitan monolaurate, triethanolamine oleate, etc. Actual methods of preparing such dosage forms are known, or will be apparent, to those skilled in this art; for example, see Remington's Pharmaceutical Sciences, Mack Publishing Company, Easton, Pennsylvania, 15th Edition, 1975.

K. Quantifier of Similarity for Electron Density Maps

As discussed in Section B above, the linear correlation coefficient, CC, can be used to quantify the degree of similarity between two electron density maps.

In general terms, therefore, we provide a method for comparing two molecular structures comprising the steps of:

providing respective first and second electron density maps for the molecular structures,

transforming one or both of the maps so that the two maps are in maximum coincidence with each other, and

quantifying the degree of correlation between the coinciding maps.

Preferably, the degree of correlation is quantified by calculating the CC for the coinciding maps. A mask may be applied to the maps before the quantification step to prevent e.g. solvent molecules from contributing to the degree of correlation. Either or both of the electron density maps may be determined experimentally, e.g. by X-ray crystallographic analysis. Alternatively or additionally, either or both may be calculated e.g. from atomic coordinate data.

The use of the CC has been tested for three structural families (i.e. three different molecular types). Within each family a number of different sets of atomic coordinates were provided. Each set varied from the other sets by an r.m.s.d. of up to about 1.8 Å. Electron density maps were computed for each atomic coordinate set. The aim was to confirm that the CC determined for each pair of maps correlated with the r.m.s.d. value for the corresponding pair of atomic coordinate sets (both within and across families). A number of CCP4 (Collaborative Computational Project 4. The CCP4 Suite: Programs for Protein Crystallography, *Acta Crystallographica*, D50, (1994), 760-763.), Unix and specially developed programs were used to perform the test. The specially developed programs are provided in Annexes 1 to 4. Annexes 5 and 6 provided respective subroutines used by the programs of Annexes 1 to 4.

In order to perform the test, the first step was to compute, for each set of atomic coordinates, the asymmetric unit of the electron density map on a relatively fine grid (e.g. 1/6th of the minimum d-spacing). This was accomplished with weighted $2F_o$-F_c coefficients using the CCP4 FFT program.

Next, for each molecule in the asymmetric unit, the atomic coordinates of the molecule were extracted from the complete coordinate set using Unix GREP. Using Astex-EXTENDC (see Annex 3), the electron density map was then extended to cover the molecule thus extracted, including a minimum 3 Å border to ensure that no parts of electron density for any atom of the molecule were unintentionally excluded.

The extracted atomic coordinates were also used to generate a molecular mask with the CCP4 NCSMASK program. Such a mask is a 3D array of grid points wherein each grid point covered by the molecule is labelled '1' and each grid point outside the molecule is labelled '0'. The coverage of the molecule was determined by 2 Å radius spheres centred on each atomic position.

Using the Astex-KFIT program (see Annex 4), each set of atomic coordinates was superposed onto a common reference and the rigid-body transformation was determined for r.m.s.d. minimisation between each pair of molecules. Each transformation was then applied, using Astex-ROTMAP (Annex 1), to the corresponding pair of electron density maps and associated masks, interpolating maps and masks onto a common unit cell and grid (e.g. at ¼ of the minimum d-spacing). The masks were interpolated linearly, whereas the electron densities were interpolated using quadratic functions.

Finally, the CCs for the pairs of transformed and interpolated electron density maps were calculated using Astex-DENCOR (Annex 2). The transformed and interpolated masks were used to ensure that only electron densities covered by the molecules contributed to the CCs.

A graph of calculated CC against calculated r.m.s.d. was plotted and the graph points fitted to the straight line $y = 1 - x/2$. This line is constrained to pass through the point $(x,y) = (0,1)$ because for zero r.m.s.d. perfect correlation is expected. The graph demonstrated that CC was strongly anticorrelated with r.m.s.d., and a linear relationship $y = 1 - x/2$ where x = r.m.s.d. and y = CC was observed. Thus an r.m.s.d. of 1.5 Å corresponds approximately to a CC of 0.25. The equation implies that for an r.m.s.d. of 2 Å or greater, no correlation of the electron densities is expected. As expected, the r.m.s.d. distances were significantly lower for pairs of molecules within the same structural family then for those taken from different families, and consequently CCs for pairs of molecules within a structural family were consistently higher than those taken from different families.

The invention is illustrated by the following example:

Example

Cloning of 3A4

3A4 corresponding to M18907 (GI_181373) was cloned from human liver library (Origene Technologies, Inc.).

PCR carried out as recommended by the manufacturer:

Liver library	2.0 µl

10X PCR buffer (-Mg^{2+})	2.5 µl
10mM dNTPs	0.5 µl
10mM $MgSO_4$	2.5 µl
Water	11.0 µl
Primer 1 (@10 pmol/µl)	3.0 µl
Primer 2 (@10 pmol/µl)	3.0 µl

Primer 1 is complementary to the 5' end of the full length 3A4 cDNA. Primer 2 is complementary to the 3' end of the cDNA and adds a four histidine tag onto the C-terminus of the 3A4 protein.

Heat to 94°C, add 0.5µl (1 Unit) Vent polymerase

35 cycles as follows:

94 °C	30 seconds
65 °C	60 seconds
72 °C	60 seconds

1 cycle of 72 °C for 5 minutes.

Following the addition of 1 µl (2.5 Units) Taq polymerase and incubation at 72 °C for 10 minutes, 1 µl of product was used in a TOPO cloning reaction (vector pCR4TOPO, Invitrogen). The cloning reaction was used to transform *E. coli* XL1-blue and positive clones identified by NdeI/SalI restriction digestion of purified plasmids. Positive clones were sequenced fully on both strands and the NdeI/SalI insert subcloned into pET20b to yield the template clone. This clone was used as the template in subsequent PCR reactions.

N-Terminal truncation of 3A4

The expression vector pCWOri+, provided by Prof. F. W. Dahlquist, University of Oregon, Eugene, Oregon, USA, was used to express the truncated human cytochrome P450 in the *E. coli* strain XL1 Blue (Stratagene). Full-length cDNA encoding cytochrome P450 3A4 isolated above was used as a template for PCR amplification, engineering the 5' terminus and insertion of a four Histidine tag at the C-terminus.

N-terminal truncation of 3A4 was carried out by PCR as outlined below, to generate the published NF10 N-terminal truncation described by Gillam (Gillam et al, Arch. Biochem. Biophys. Vol. 305, 123-131, 1993).

Template		~5 ng
10X PCR buffer (+Mg^{2+})		5.0 µl
10mM dNTPs		1.0 µl
Water		42.0 µl
Primer 2	(@100 pmol/µl)	0.5 µl
Primer 3	(@100 pmol/µl)	0.5 µl
Vent polymerase (2 U/µl)		0.5 µl

25 cycles of:

94 °C	30 seconds
65 °C	60 seconds
72 °C	60 seconds

1 cycle of 72 °C for 5 minutes.

Following the addition of 1 µl (2.5 units) Taq polymerase and incubation at 72 °C for 10 minutes, 1µl of product was used in a TOPO cloning reaction (vector pCR4TOPO, Invitrogen). The cloning reaction was used to transform *E. coli* XL1-blue and positive clones identified by NdeI/SalI restriction digestion of purified plasmids. Positive clones were sequenced fully and the NdeI/SalI insert subcloned into pCWori+ to yield clone p3A4. This clone was used for protein expression.

Primer 1

5’-GGAATTCATATGGCTCTCATCCCAGACTTGGCC-3’

Primer 2

5’-TGCGGTCGACTCAATGGTGATGGTGGGCTCCACTTACGGTGCCATCC-3’

Primer 3

5’-TTAACATATGGCATATGGTACTCATTCACATGGTCTGTTTAAAAACTGGGAATTCCAGGGCCCACACC-3’

Bacterial expression

A single ampicillin resistant colony of XL1 blue cells was grown overnight at 37 ˚C in Terrific Broth (TB) with shaking to near saturation and used to inoculate fresh TB media. Bacteria were grown to an OD600nm =0.5 in 1 litre of TB broth containing 100 µg/ml of ampicillin at 37 °C at 185 rpm in 2 litre flask. The haem precursor delta aminolevulinic acid (80 mg/l) was added 30 min prior to induction with 1 mM isopropyl-β-D-thiogalactopyranoside (IPTG) and the temperature lowered to 25 °C. The bacterial culture was continued under agitation at 25 °C for 48 hours.

Protein purification 1A

Cells expressing 3A4 grown as described above were pelleted at 10000 g for 10 min and resuspended in a buffer containing 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 0.1% (v/v) of protease inhibitor cocktail (Calbiochem), 10 mM imidazole, 40U/ml DNase 1 and 5 mM $MgSO_4$.

The cells were lysed by passing twice through a Constant Systems Cell Homogeniser at 10000 psi. The cell debris was then removed by centrifugation at 22000 x g at 4 °C for 30 min.

Detergent IGEPAL CA630 (Sigma) was added dropwise from a 10% stock solution to the lysate at a final concentration of 0.3% (v/v) and the lysate was incubated with previously washed NiNTA resin (Qiagen) overnight at 4 °C, using agitation. The protein bound-NiNTA resin was pelleted by centrifugation at 2000 g for 2 min at 4 °C. The resin was washed with 20 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 10 mM imidazole, 1:1000 dilution of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630 and the resin pelleted by centrifugation at 2000 xg for 2 min at 4 °C. The resin was then washed with 10 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 20 mM imidazole, 0.1% (v/v) protease inhibitors, 0.3% IGEPAL CA630 and the resin recovered by centrifugation as described above.

The resin was packed into a column at 4 °C and the cytochrome P450 eluted with 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 300 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630.

The cytochrome P450 obtained from the NiNTA column was quickly desalted into 10 mM KPi, pH 7.4, 20% glycerol, 2.0 mM DTT, 1 mM EDTA using a HiPrep 26/10 desalting column (Pharmacia), at a flow rate of 5 ml/min.

The desalted cytochrome P450 was directly applied to a CM Sepharose column (Pharmacia), previously equilibrated with 10 mM KPi, pH 7.4, 20% glycerol, 2.0 mM DTT, 1 mM EDTA. The following step elution was applied: wash with 20 column volumes of 10 mM KPi, pH 7.4, 20% glycerol, 2.0 mM DTT, 1 mM EDTA, wash with the above buffer with 75 mM KCl in order to remove any trace of detergent, then eluted with the above buffer with KCl concentration increased to 500 mM.

The protein was concentrated up to 40 mg/ml using a microconcentrator for crystallization assays.

Protein Characterization

The quality of the final preparation was evaluated by:

(a) SDS polyacrylamide gel electrophoresis: This was performed using commercial gels (Nugen) followed by CBB staining according to the manufacturer's instructions. The purity as estimated by scanning a digital image of a gel was estimated to be at least 95%.

(b) Mass Spectroscopy: This was performed using a Bruker "BioTOF" electrospray time of flight instrument. Samples were either diluted by a factor of 1000 straight from storage buffer into methanol/water/formic acid (50:48:2 v/v/v), or subjected to reverse phase HPLC separation using a C4 column.

Calibration was achieved using Bombesin and angiotensin I using the 2+ and 1+ charged states. Data were acquired between 200 and 2000*m/z* range and were subsequently processed using Bruker's X-mass program. Mass accuracy was typically below 1 in 10 000.

Mass spec of 3A4: 55281 Da(observed)
55278 Da (predicted minus N-terminal methionine)

<u>Crystallization 1A</u>

Crystals of the 3A4 were grown using the hanging drop vapor diffusion method. Protein at 40 mg/ml in 10mM Kpi pH 7.4, 0.5 M KCl, 2mM DTT, 1mM EDTA. 20% glycerol, was mixed in a 1:1 ratio, using 0.5ul drops, with a reservoir solution. The crystals of 3A4 grew over a reservoir solution containing 0.1 M HEPES pH 7.5, 0.2 M sodium chloride, 30% PEG 400.

Alterative conditions are listed below:
0.1 M HEPES pH 7.5, 0.2 M sodium chloride, 30% PEG 400
0.05 M HEPES pH 7.5, 0.2 M sodium chloride, 35% PEG 400
0.05 M HEPES pH 7.5, 0.2 M sodium chloride, 30% PEG 400
0.15 M Imidazole-HCl pH 8, 10% 2-propanol
0.1 M 2-(N-cyclohexylamino)ethanesulfonic acid (CHES) pH 9.5, 30% PEG 400
0.15 M Hepes - Na pH 7.5, 5% IPA, 10% Peg 4000
0.1 M phosphate-citrate pH 4.2, 1.6 M NaH2PO4/ 0.4M K2HPO4
0.1 M citrate pH 5.5, 0.2 sodium chloride, 1.0 M Ammonium phosphate
0.2 M Lithium chloride, 20% PEG 3350
0.2 M Potassium chloride , 20% PEG 3350
0.2 M Sodium formate , 20% PEG 3350
0.2 M Potassium formate , 20% PEG 3350
0.2 M Ammonium formate , 20% PEG 3350
0.2 M Lithium acetate, 20% PEG 3350

0.2 M Potassium chloride, 20% PEG 3350
0.2 M Sodium formate , 20% PEG 3350
0.2 M Lithium acetate , 20% PEG 3350
0.2 M Sodium acetate , 20% PEG 3350
0.2 M Potassium acetate , 20% PEG 3350
0.2 M Ammonium acetate, 20% PEG 3350
0 .1 M HEPES pH 7.5, 0.2 M sodium chloride , 30% PEG 400
0.1 M HEPES pH 7.5, 5% Iso-Propanol, 10% PEG 4000
200 mM K Acetate, 25% peg 3350
200 mM K Acetate , 25% peg 3350
300 mM Na acetate, 25% peg 3350
200 mM Sodium formate, 25% PEG 3350
0. 300 M Lithium acetate, 25.0 % PEG 3350
0. 100 M Imidazole-HCl pH 8, 10% 2-propanol
0. 150 M Imidazole-HCl pH 8, 10% 2-propanol
Crystals formed within 1-7 days at 25 °C, and were rod shaped in morphology.

The approximate cell dimensions of the crystals were a=77 Å, b=99 Å, c=129 Å, β=90 °. The space group is I222.

The crystals were flash frozen in liquid nitrogen, using 80% reservoir solution, 20% ethylene glycol as a cryoprotectant.

Crystals of 3A4 were also grown over a reservoir solution containing:
0.15M HEPES pH7.5, 5% IPA, 10 % PEG 4000.

Crystals were obtained with unit cell C2: a=152Å, b=101 Å, c=78Å, α=90°, β=120°, γ=90°. The invention thus provides crystal of 3A4 having this space group and unit cell dimensions, the dimensions a, b and c and β varying independently by +/- 5%.

In summary the invention includes a crystal of 3A4 having a space group I222 and unit cell size a=77 Å, b=99 Å, c=129 Å, β=90°; or having a space group C2 and unit cell size a=152Å, b=101 Å, c=78Å, α=90°, β=120°, γ=90°. Those of skill in the art will recognise that the cell dimensions of the crystal may vary by 5%, though preferably by 1-2Å, upon repeat crystallization, and such variation resides within the spirit and scope of the invention.

Protein purification (1B)

The cells were pelleted at 10000 g for 10 min and resuspended in a buffer containing 500 mM KPi, pH 7.4, 20 % glycerol (v/v), 10 mM mercaptoethanol, 0.1% (v/v) of protease inhibitor cocktail 3 (Calbiochem), 10 mM imidazole, 40U/ml DNase 1 and 5 mM $MgSO_4$.

Passing twice through a Constant Systems Cell Homogeniser at 10000 psi lysed the cells. The cell debris was then removed by centrifugation at 22000 x g at 4 °C for 30 min.

Detergent IGEPAL CA630 (Sigma) was added dropwise from a 10% stock solution to the lysate at a final concentration of 0.3% (v/v) and the lysate was incubated with previously washed NiNTA resin (Qiagen) overnight at 4 °C, using agitation. The protein bound-NiNTA resin was pelleted by centrifugation at 2000 g for 5 min at 4 °C. The resin was washed with 20 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 10 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630 and the resin pelleted by centrifugation at 2000 g for 5 min at 4 °C. The resin was then washed with 10 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 20 mM imidazole, 0.1% (v/v) protease inhibitors, 0.3% IGEPAL CA630 and the resin recovered by centrifugation as described above.

The resin was packed into a column at room temperature and the cytochrome P450 eluted with cold 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 300 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630.

The cytochrome P450 obtained from the NiNTA column was quickly desalted into 20 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA using a HiPrep 26/10 desalting column (Pharmacia), at a flow rate of 5 ml/min on a Akta FPLC system (Pharmacia). A watch UV command (280 nm) of greater than 750 mAu was then used to divert the desalted P450 from the HiPrep 26/10 desalting column onto a CM Sepharose column (Pharmacia), previously equilibrated with 20 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA for final purification. The peak divert was ended when the mAu fell below 750mAu. The following step elution was then applied to the CM Sepharose column: wash with 10 column volumes of 20 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA, followed by a wash with 6 column volumes with the above buffer with 75 mM KCl added in order to remove any trace of detergent, then eluted with the above buffer with KCl concentration increased to 500 mM.

The protein was concentrated up to 40 mg/ml using a microconcentrator for crystallization trials.

Crystallization (1B)

Crystals of the 3A4 were grown using the hanging drop vapour diffusion method. Protein at 37.4 mg/ml in 20 mM Kpi pH 7.2, 0.5 M KCl, 2mM DTT, 1mM EDTA, 20% glycerol, was mixed in a 1:1 ratio, using 0.5ul drops, with a reservoir solution. The crystals of 3A4 grew over a reservoir solution containing 0.15 M HEPES pH 7.5, 2.5% IPA, 10% PEG 4000.

Crystals formed within 1-7 days at 25 °C, and were rod shaped in morphology.

The crystals were flash frozen in liquid nitrogen, using crystallisation solution supplemented with 15% glycerol as a cryoprotectant.

Dataset collection (1)

A native dataset was collected at the ESRF beamline 14.2 to a resolution of 2.7 Å, from a crystal produced using the protocol above in Protein purification (1B) and Crystallisation (1B).

The cell dimensions of the crystals were a=77.85 Å, b=99.71 Å, c=132.74 Å, α=β=γ=90 °. The space group was I222.

A total of 100 one degree oscillation images were collected, processed with MOSFLM (Leslie, A. G. W. (1992). In *Joint CCP4 and EESF-EACMB Newsletter on Protein Crystallography*, vol. 26, Warrington, Daresbury Laboratory), scaled using SCALA (CCP4 – Collaborative Computational Project 4. (1994) The CCP4 Suite: Programs for Protein Crystallography. *Acta Crystallographica* D50, 760-763) and reduced using the CCP4 suite of programs.

Protein purification (2)

The cells were pelleted at 10000 g for 10 min and resuspended in a buffer containing 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 0.1% (v/v) of protease inhibitor cocktail 3 (Calbiochem), 10 mM imidazole, 40U/ml DNase 1 and 5 mM $MgSO_4$.

Passing twice through a Constant Systems Cell Homogeniser at 10000 psi lysed the cells. The cell debris was then removed by centrifugation at 22000 g at 4 °C for 30 min.

Detergent IGEPAL CA630 (Sigma) was added dropwise from a 10% stock solution to the lysate at a final concentration of 0.3% (v/v) and the lysate was incubated with previously washed NiNTA resin (Qiagen) overnight at 4 °C, using agitation. The protein bound-NiNTA resin was pelleted by centrifugation at 2000 g for 5 min at 4 °C. The resin was washed with 20 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 10 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630 and the resin pelleted by centrifugation at 2000 g for 5 min at 4 °C. The resin was then washed with 10 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 20 mM imidazole, 0.1% (v/v) protease inhibitors, 0.3% IGEPAL CA630 and the resin recovered by centrifugation as described above.

The resin was packed into a column at room temperature and the cytochrome P450 eluted with cold 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 300 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630.

The cytochrome P450 obtained from the NiNTA column was quickly desalted into 10 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA, 10mM K_2SO_4 using a HiPrep 26/10 desalting column (Pharmacia), at a flow rate of 5 ml/min.

The desalted cytochrome P450 was directly applied to a CM Sepharose column (Pharmacia) previously equilibrated with 10 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA, 10mM K_2SO_4. The following step elution was applied: wash with 20 column volumes of 10 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA, 10mM K_2SO_4 followed by a wash with 20 column volumes of the above buffer with 75 mM KCl in order to remove any trace of detergent, then eluted with the above buffer with KCl concentration increased to 500 mM.

The protein was concentrated up to 20 mg/ml using a microconcentrator for crystallization assays.

Crystallization (2)

Crystals of the 3A4 were grown using the hanging drop vapour diffusion method. Protein at 18.5 mg/ml in 10 mM Kpi pH 7.2, 0.5 M KCl, 2 mM DTT, 1 mM EDTA, 20% glycerol, 10 mM K2SO4 was mixed in a 1:1 ratio, using 0.5ul drops, with a reservoir solution. The crystals of 3A4 grew over a reservoir solution containing 0.1 M HEPES pH 7.2, 5% IPA, 10% PEG 4000. The crystal was frozen using the crystallization solution supplemented by glycerol to 33%.

Crystals formed within 1-7 days at 25 °C, and were rod shaped in morphology.

Dataset collection (2)

A native dataset was collected at the ESRF beamline 14.2 to a resolution of 2.8 Å, from a crystal produced using the protocol above in Protein purification (2) and Crystallisation (2).

The approximate cell dimensions of the crystals were a= 77.32 Å, b=100.37 Å, c=132.72 Å, $\alpha=\beta=\gamma=90$ °. The space group was I222.

A total of eighty one degree oscillation images were collected, processed with MOSFLM (Leslie, A. G. W. (1992). In *Joint CCP4 and EESF-EACMB Newsletter on Protein Crystallography*, vol. 26, Warrington, Daresbury Laboratory), scaled using SCALA (CCP4 – Collaborative Computational Project 4. (1994) The CCP4 Suite: Programs for Protein Crystallography, *Acta Crystallographica* D50, 760-763) and reduced using the CCP4 suite of programs.

Protein purification (3)

The cells were pelleted at 10000 g for 10 min and resuspended in a buffer containing 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 0.1% (v/v) of protease inhibitor cocktail 3 (Calbiochem), 10 mM imidazole, 40U/ml DNase 1 and 5 mM $MgSO_4$.

Passing twice through a Constant Systems Cell Homogeniser at 10000 psi lysed the cells. The cell debris was then removed by centrifugation at 22000 x g at 4 °C for 30 min.

Detergent IGEPAL CA630 (Sigma) was added dropwise from a 10% stock solution to the lysate at a final concentration of 0.3% (v/v) and the lysate was incubated with previously washed NiNTA resin (Qiagen) overnight at 4 °C, using agitation. The protein bound-NiNTA resin was pelleted by centrifugation, 2000 g for 5 min at 4 °C. The resin was washed with 20 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 10 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630 and the resin pelleted by centrifugation at 2000 g for 5 min at 4 °C. The resin was then washed with 10 resin volumes of 500 mM KPi, pH 7.4, 20% glycerol, 10 mM mercaptoethanol, 20 mM imidazole, 0.1% (v/v) protease inhibitors, 0.3% IGEPAL CA630 and the resin recovered by centrifugation as described above.

The resin was packed into a column at room temperature and the cytochrome P450 eluted with cold 500 mM KPi, pH 7.4, 20 % glycerol, 10 mM mercaptoethanol, 300 mM imidazole, 0.1% (v/v) of protease inhibitor cocktail, 0.3%(v/v) IGEPAL CA630.

The cytochrome P450 obtained from the NiNTA column was quickly desalted into 10 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA using a HiPrep 26/10 desalting column (Pharmacia), at a flow rate of 5 ml/min.

The desalted cytochrome P450 was directly applied to a CM Sepharose column (Pharmacia) previously equilibrated with 10 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA. The following step elution was applied: wash with 20 column volumes of 10 mM KPi, pH 7.2, 20% glycerol, 2.0 mM DTT, 1 mM EDTA, followed by a wash with 20 column volumes of the above buffer with 75 mM KCl in order to remove any trace of detergent, then eluted with the above buffer with KCl concentration increased to 500 mM.

The concentrated sample (200 µL, 7.9 mg protein) was then gel filtered using a Superdex 200 HR10/30 column (Pharmacia) in 10 mM KPi, pH7.2, 20% glycerol, 1 mM EDTA, 2 mM DTT, 500 mM KCl at a flow rate of 0.4 ml/min. Fractions of 0.5 ml were collected. Three peaks of protein were collected, of these the first (elution volume, Ve = 8.64 ml) represented aggregated protein that had been excluded by the void volume, Vo (Vo = 8.66 ml) of the column, the second peak (Ve = 12.4 ml) was the largest and represented the P450, and the third and smallest peak (Ve = 15.49 ml) was low molecular weight protein contaminants.

The P450 peak was then pooled and concentrated up to 40 mg/ml using a microconcentrator for crystallization trials. 3A4 can alternatively be purified by gel filtration chromatography, by passage down a 26/60 Superdex 200 column equilibrated in 10mM K Pi pH 7.2, 20% glycerol, 0.5M KCl, 2mM DTT run at 1.5mg/ml, to improve homogeneity for crystallisation.

Crystallization (3)

Crystals of the 3A4 were grown using the hanging drop vapour diffusion method. Protein at 36 mg/ml in 10 mM Kpi pH 7.2, 0.5 M KCl, 2 mM DTT, 1 mM EDTA, 20% glycerol, was mixed in a 1:1 ratio, using 0.5 μl drops, with a reservoir solution. The crystals of 3A4 grew over a reservoir solution containing 0.1 M HEPES pH 7.5, 0.025 M sodium chloride, 7.5% IPA, 10% PEG 4000.

The crystals formed over a number of days at 25°C, and were rod shaped in morphology.

The crystals were transferred to a cryo-solution consisting of 0.1 M HEPES pH 7.5, 0.25 M KCl, 15% PEG 4000 and 20% glycerol and then frozen in liquid nitrogen prior to data collection.

Dataset collection (3)

Data was collected from a single crystal, produced using the protocol above in Protein purification (3) and Crystallisation (3), at beamline ID29 at the European Synchrotron Radiation Facility to a resolution of 2.8 Å. An energy scan was taken from the crystal prior to data collection to determine the precise energy at which the haem iron provided a detectable signal. The energy scan indicated the peak energy to be 7.126 KeV (corresponding to a wavelength of 1.7398 Å), and a suitable point of inflection wavelength to be 7.123 KeV (corresponding to a wavelength of 1.7406 Å).

The approximate cell dimensions of the crystals were a=77.94 Å, b=100.91 Å, c=131.00 Å, $\alpha=\beta=\gamma=90$ °. The space group was I222.

Two datasets were collected from a single crystal, one at a wavelength of 1.7398 Å (peak dataset) to a resolution of 2.8 Å and the second at a wavelength of 1.7406 Å (inflection dataset) to a resolution of 3.1 Å. A total of 180° of data were collected at each wavelength to ensure that the data were redundant. The data were processed using MOSFLM (Leslie, A. G. W. (1992). In *Joint CCP4 and EESF-EACMB Newsletter on Protein Crystallography*, vol. 26, Warrington, Daresbury Laboratory), scaled using SCALA (CCP4 computing package (Collaborative Computational Project 4. The CCP4 Suite: Programs for Protein Crystallography, *Acta*

Crystallographica, D50, (1994), 760-763) and further reduced using the CCP4 suite of programs.

Table 1 below contains the data statistics for the peak wavelength data.
Table 1: Data statistics

Dmax (Å)	Dmin(Å)	Rmerge	Rfull	Rcum	Ranom	I/sigma	Mn(I)/sd
50.00	8.85	0.052	0.043	0.052	0.034	9.2	41.5
8.85	6.26	0.044	0.037	0.048	0.031	9.4	37.9
6.26	5.11	0.045	0.039	0.048	0.029	14.7	34.9
5.11	4.43	0.047	0.038	0.047	0.023	13.7	34.2
4.43	3.96	0.054	0.044	0.049	0.027	12.3	29.4
3.96	3.61	0.082	0.069	0.053	0.035	8.0	21.3
3.61	3.35	0.135	0.112	0.058	0.060	5.1	12.4
3.35	3.13	0.221	0.180	0.064	0.095	3.3	7.6
3.13	2.95	0.380	0.280	0.069	0.193	1.9	3.9
2.95	2.80	0.626	0.430	0.073	0.352	1.2	2.0
Overall:		0.073	0.059	0.073	0.049	6.6	18.8

Where:
Dmax = minimum resolution
Dmin = maximum resolution

```
        sum~i~ ( sum~j~ | I~j~ - <I> | )
Rmerge = -------------------------------
            sum~i~ ( sum~j~ <I> )
```

I~j~ = the intensity of the jth observation of reflection i
<I> = the mean of the intensities of all observations of
reflection I
sum~i~ is taken over all reflections
sum~j~ is taken over all observations of each reflection.

Rfull = Rmerge for fully recorded reflections only
Rcum = cumulative Rmerge for all reflection
Ranom = Sum |Mn(I+) - Mn(I-)| / Sum (Mn(I+) + Mn(I-)), where MN(I) is the mean I of that shell.
I/sigma = I/Sigma
Mn(I)/Sd = Mn(I)/standard deviation of I/Sigma(I)

MAD structure determination

The location of the iron atom within the unit cell was determined by visual inspection of the three Harker sections of the anomalous difference Patterson map calculated using the peak anomalous data by the program FFT (part of the CCP4 suite).

The refined parameters of the iron atom used to generated phases are as follows: x= 23.255, y=23.237, z=10.742, occupancy=0.92, temperature factor=69.45. These refined parameters were obtained using the program SHARP, by refinement against the experimental data obtained from the crystal (columns 4 and 5 of Table 3). These atom parameters were then used within SHARP to generate phases for 3A4. These phases can then be modified by density modification procedures. The phases from SHARP were solvent flattened using SOLOMON/DM as available through the SHARP program. The resulting solvent flattened structure factor amplitudes and phases are given in columns 6 and 7 of Table 3.

We choose to refine the iron atom parameters within SHARP, generate phases within SHARP and then perform density modification using SOLOMON and DM as implemented through SHARP. It however would be possible to generate phases using the heavy atom parameters given above and to solvent flatten the resulting phases using alternative programs (for example using the CCP4 program MLPHARE ((Z.Otwinowski: Daresbury Study Weekend proceedings, 1991) to generate the phases and the CCP4 program DM (K. Cowtan (1994), Joint CCP4 and ESF-EACBM Newsletter on Protein Crystallography, 31, p34-38).

The generation of such phases (unflattened or solvent flattened) is reliant on determining accurate parameters that describe the heavy or anomalous atom (in this case the iron of the haem), as are given above.

Thus in a further aspect the invention provides a method of generating phases of crystals of 3A4 using the iron parameters x= 23.255, y=23.237, z=10.742, occupancy=0.92, temperature factor=69.45 and the experimental structure factor data obtained from the crystal (columns 4 and 5 of Table 3) or structure factor data obtained from another crystal of 3A4 in the same crystal form.

This assignment of the iron position was consistent with the given space group I222 and not with the alternative choice $I2_12_12_1$. Both datasets together with the spacegroup I222 were giving to the program autoSHARP (Vonrhein, C. & Bricogne, G., (2002), Global Phasing) that automatically determined the position and handedness of the heavy atom substructure solution, resulting in a set of phases after density modification. The resulting density modified phases were used as phase restraints during further refinement of the heavy atom model in SHARP (La Fortelle, E. de and Bricogne, G. (1997). Maximum-likelihood heavy-atom parameter refinement for multiple isomorphous replacement and multiwavelength anomalous diffraction methods. *Methods in Enzymology* **276,** 472-494) to give a set of phases (phase set I). In a similar heavy atom refinement and phasing experiment, using the peak wavelength alone, a set of phases (phase set II) was obtained.

The resulting phases (phase set I) were used in phased molecular replacement as implemented in MOLREP (A.Vagin, A.Teplyakov, J. Appl. Cryst. (1997) 30, 1022-1025, part of the CCP4 suite) and using 2C5 with the haem excluded (pdbent 1DT6) as a search model together with the sequence of SEQ ID 2. This gave an unambiguous solution where the haem moiety was consistent with the iron position obtained through inspection of the Harker sections.

The oriented and positioned model (based on 1DT6 and the sequence of SEQ ID 2), model-A, was used together with the phase set II phases in density modification as implement in SOLOMON (Abrahams J. P. and Leslie A. G. W., *Acta Crystallographica* **D52,** (1996), 30-42) through the SHARP program package.

Table 2 contains the phasing statistics in resolution bins. The columns are:
Minimum resolution
Maximum resolution
Number of acentric reflections for peak and inflection dataset used in phasing power calculation
Anomalous phasing power for peak and inflection dataset (phase set I)
Number of acentric reflections used in SHARP figure-of-merit calculation
SHARP figure-of-merit for acentric reflections (phase set I)
Number of centric reflections used in SHARP figure-of-merit calculation
SHARP figure-of-merit for centric reflections (phase set I)
Figure-of-merit after density modification of phase set II with SOLOMON including model-A at the very first cycle – final cycle of solvent flipping.
10. Average phase error derived from the FOM given in column 9 using the relationship FOM=cos(average phase error)

Structure factors and phases from which the electron density map can be calculated are contained in Table 3. The resulting electron density map showed clear structural features. When comparing the electron density with the molecular replacement solution, the secondary structure of P450 was apparent, although structural elements were clearly slightly displaced from their location in the 2C5 search model. The haem group, missing from the molecular replacement model, has clearly defined planar electron density.

Protein Characterization

The final quality of each of the protein preparations was evaluated by:

(a) SDS polyacrylamide gel electrophoresis

This was performed using commercial gels (Nugen) followed by coomassie brilliant blue (CBB) staining according to the manufacturer's instructions. The purity as estimated by scanning a digital image of a gel was estimated to be at least 95%.

(b) Mass Spectroscopy

Mass spectrometry was performed using a Bruker BioTOF II electrospray time of flight instrument. Samples were either diluted by a factor of 1000 straight from storage buffer into methanol/water/formic acid (50:48:2 v/v/v), or subjected to a reverse phase separation using a C4 Millipore 'zip-tip' or a C4 HPLC column, before being diluted into methanol/water/formic acid.

Calibration was achieved by measurement of the 2+ and 1+ charge states of a peptide mixture containing Bombesin and angiotensin I or by using the multiple charge states of Horse Myoglobin. Data were acquired in the *m/z* range 200 to 2000 and were subsequently processed using Bruker's X-mass program. Mass accuracy was expected to be better than 1 in 10 000 (100ppm).

Mass spec of 3A4: 55279.43 Da (observed)
55277.81 Da (predicted for protein minus the N-terminus Methionine)

(c) Functionality assays

Activity assays on 3A4 were performed using dibenzylfluorescein (Gentest), which is dealkylated to the fluorescein ester, as a fluorescent substrate.

Assays were carried out in 96-well half-area black, Costar plates in a final assay volume of 50 µl. The reaction rates were monitored for 1 hour at room temperature on a Fluoroscan Ascent FL Instruments (Labsystem) platereader with excitation and emission wavelengths of 485 nm and 538 nm respectively. Reaction rates were measured using Prizm (GraphPad) software

Reaction mixtures were composed of 300 nM of 3A4 enzyme incubated with 2 units/ml purified human oxidoreductase, 2.8 µM dibenzylfluorescein and a regeneration system composed of 140 µM NADP$^+$, 400 µM glucose-6-phosphate and 2.8 units/ml glucose-6-phosphate dehydrogenase in 100 mM potassium phosphate pH 7.8, 1 mM $MgCl_2$.

Table 2: Phasing statistics in resolution bins

Dmin	Dmax	Nacen peak/infl	PP_acen peak/infl	Nacen	FOMacen	Ncen	FOMcen	FOM denmod	Average phase error
66.98	12.32	107/104	2.593/1.151	108	0.57611	75	0.14648	0.75897	40.6°
12.32	8.78	206/195	1.813/1.459	207	0.52620	72	0.07944	0.89841	26.1°
8.78	7.19	287/275	2.437/1.473	288	0.56634	74	0.08138	0.90077	25.7°
7.19	6.24	348/333	2.606/1.648	349	0.53477	75	0.07990	0.83914	33.0°
6.24	5.58	396/386	2.420/1.434	396	0.53154	62	0.10538	0.85947	30.7°
5.58	5.10	440/436	2.000/1.087	440	0.48298	77	0.07540	0.84261	32.6°
5.10	4.72	491/489	1.658/0.863	492	0.45701	77	0.05287	0.86452	30.2°
4.72	4.42	532/530	1.477/0.666	532	0.41346	65	0.04342	0.86050	30.6°
4.42	4.17	557/555	1.180/0.488	557	0.38258	79	0.04054	0.85383	31.4°
4.17	3.95	594/593	0.994/0.429	595	0.36431	60	0.03930	0.84404	32.4°
3.95	3.77	615/614	0.763/0.290	618	0.30101	77	0.04647	0.81426	35.5°
3.77	3.61	656/653	0.611/0.219	659	0.24232	77	0.03926	0.83271	33.6°
3.61	3.47	679/676	0.474/0.189	682	0.19202	75	0.04716	0.77315	39.4°
3.47	3.34	709/707	0.420/0.156	715	0.16313	72	0.05622	0.73713	42.5°
3.34	3.23	738/734	0.340/0.141	740	0.13450	73	0.04334	0.70163	45.4°
3.23	3.13	757/305	0.299/0.132	763	0.11239	78	0.04350	0.64421	49.9°
3.13	3.04	772/0	0.241/0.000	785	0.09334	69	0.05388	0.59097	53.8°
3.04	2.95	716/0	0.205/0.000	781	0.08119	40	0.06355	0.52079	58.6°

2.95	2.87	626/0	0.189/0.000	689	0.08130	34	0.05063	0.38949	67.1°
2.87	2.80	563/0	0.167/0.000	617	0.07522	22	0.05246	0.30668	72.1°
Total		10789/7585	0.779/0.446	11013	0.25754	1333	0.06256	0.70319	45.3°

3A4 Structure Determination.

The electron density map, described by Table 3 allowed a model of 3A4 to be built using the graphical program O (Jones, T. A., Zou, J. Y., Cowan, S. W., and Kjeldgaard (1991) *Acta Cryst. A47*, 110-119). This model was then refined to 2.8Å resolution against the peak wavelength dataset from the iron MAD experiment (statistics of the data given in Table 1) using the programs CNX (Brünger, A. T., Adams, P. D., Clore, G. M., DeLano, W. L., Gros, P., Grosse-Kunstleve, R. W., Jiang, J. S., Kuszewski, J., Nilges, M., Pannu, N. S., Read, R. J., Rice, L. M., Simonson, T., and Warren, G. L. (1998) *Acta Cryst. D54*, 905-921) and Refmac (Murshudov, G. N., Vagin, A. A., and Dodson, E. J. (1997) *Acta Cryst. D50*, 760-763). The refinement statistics in Table 4 are of the model given in Table 5. The model includes 29 ordered water molecules.

Table 4: Refinement statistics of the 3A4 crystal structure:

Resolution	2.8Å
R factor	24.36%
Free R factor (5% of data)	27.38%
r.m.s.d. bonds	0.0083Å
r.m.s.d. angles	1.904°
Average B factor (all atoms)	64Å^2

ANNEX 1

```
      PROGRAM ROTMAP
C
C#### Get mean solvent density from e.d. map & NCS mask.
C#### Rotate & translate map & mask, reset density outside mask.
C#### The rotated/translated map is interpolated using a least-squares
C#### 27-point fit of a quadratic function.
C#### Memory for map storage is allocated dynamically.
C
C#### Ian J. Tickle, Astex Technology.
C#### Copyright (c) 2003 Astex Technology Ltd.  All rights reserved.
C
C#### This works in a similar way to the CCP4-MAPROT program, but
C#### addresses a "feature" (or bug?) in that program, where the density
C#### is masked before transformation & interpolation instead of
C#### afterwards as logically it should be.  This results in a shrinkage
C#### of the masked volume and a consequent over-estimation of the
C#### correlation coefficient.
C
C#### Link with standard CCP4 libraries.
C#### Additional routine required: namelist.f .
C
C#### Usage:
C
C#### Command-line input:
C#### rotmap  rootname_input  rootname_output
C
C#### Standard input:
C#### Namelist format, i.e. KEYWORD = VALUE pairs, either comma or
C#### newline-separated.
C
C#### Keywords & associated values:
C
C#### CELL = a b c        Unit cell lengths of output map.
C#### GRID = nx ny nz     Grid divisions/unit cell for output map.
C#### NMOL = nm           No of mols/a.u.
C#### ROTA = a1 a2 a3     Eulerian rotation angles (CCP4 convention).
C#### TRAN = tx ty tz     Orthogonal translations.
C
C#### Unix example:
C
C#### rotmap  3A4-apo  3A4-apo-rotmap  <<EOD
C#### CELL=86.2437 76.1479 55.6283, GRID=128 108 80
C#### ROTA=277.20 82.08 189.97, TRAN=124.810 52.354 -18.788
C#### EOD
C
C
      IMPLICIT NONE
      CHARACTER A*80,FN*255
      INTEGER I,J,K,L,LU,LV,LW,MM,MU,MV,MW,NC,NG,NKEY,NMOL,NS
      REAL C,DR,PI,RHO0,RHOA,RHOS,S,T,VC
      PARAMETER(NKEY=13,PI=3.14159265,DR=PI/180.)
C
      CHARACTER FMT(NKEY)*8,KEYA(NKEY)*9,TALC(4),TYPE(NKEY)
      LOGICAL LDEF(NKEY),LINP(NKEY),LVAL(NKEY)
      INTEGER HDEF(NKEY),HVAL(NKEY),IDEF(NKEY),IP(3),IP1(3),IP2(3),
     &IVAL(NKEY),LALC(4),LUVW1(3),LUVW2(3),MUVW1(3),MUVW2(3),NFMT(NKEY),
     &NXYZ(3),NXYZ1(3),NXYZ2(3)
```

```
      REAL AV(3),CA(3),CCD1(6),CCD2(6),CCD(6),CCG(3),OMCD(3,3),
     &OMGD(3,3),RDEF(NKEY),RMG1(3,3),RMG2(3,3),RMO(3,3),RTM(4,4,192),
     &RVAL(NKEY),SA(3),TVG1(3),TVG2(3),TVO(3)
C
      COMMON/INPCOM/ CCG,NXYZ,NMOL,AV,TVO
      COMMON/MAPCOM/ LU,LV,LW,MU,MV,MW,NC,RHO0,VC,RMG1,RMG2,TVG1,TVG2
      EQUIVALENCE
     &(LDEF,HDEF,IDEF,RDEF),
     &(CCG,LVAL,HVAL,IVAL,RVAL),
     &(LU,LUVW1(1)),(MU,MUVW1(1))
      INTEGER LENSTR
      EXTERNAL GETDEN,PUTDEN
C
C#### Keywords & default values for parameters.
      DATA KEYA/'CELL',2*' ','GRID',2*' ','NMOL','ROTATE',2*' ',
     &'TRANSLATE',2*' '/
      DATA FMT/'3F8.3',2*' ','3I8',2*' ','I8','3F8.2',2*' ','3F8.3',
     &2*' '/
      DATA TYPE/NKEY*' '/, NFMT/NKEY*0/
      DATA CCD,CCG,NXYZ,NMOL,AV,TVO/3*0.,3*90.,3*0.,3*0,1,6*0./
      DATA IP/3,1,2/
C
      CALL CCPFYP
      IF (IARGC().NE.2) CALL CCPERR(1,
     &'Usage:  rotmap  root_name_in  root_name_out')
C
C#### Read in non-default parameter values & check.
      CALL NAMELIST(4,NKEY,KEYA,FMT,NFMT,TYPE,LDEF,HDEF,IDEF,RDEF,LINP,
     &LVAL,HVAL,IVAL,RVAL)
C
      DO I=1,3
        IF (CCG(I).LE.0. .OR. NXYZ(I).LE.0) CALL CCPERR(1,
     &  'ERROR: No defaults for CELL or GRID.')
      ENDDO
C
C#### Get root of input filenames.
      CALL GETARG(1,FN)
      L=LENSTR(FN)
C
C#### Open input CCP4 format mask and map & check map headers.
      WRITE(6,'(/)')
      CALL MAPHEAD(1,FN(:L)//'.msk',A,IP1,LUVW1,MUVW1,NXYZ1,NS,MM,CCD1,
     &RHOA,RHOS)
      IF (MM.NE.0) CALL CCPERR(1,'Map mode must be 0.')
      CALL MAPHEAD(2,FN(:L)//'.map',A,IP2,LUVW2,MUVW2,NXYZ2,NS,MM,CCD2,
     &RHOA,RHOS)
      IF (MM.NE.2) CALL CCPERR(1,'Map mode must be 2.')
C
C#### Get symmetry info.
      CALL MSYMOP(2,NG,RTM)
      NMOL=NG*NMOL
C
C#### Check that unit cells are consistent & compute input cell volume.
      NC=1
      VC=1.
      S=1.
      T=2.
      DO I=1,3
        IF (ABS(CCD2(I)-CCD1(I)).GT.1E-5*CCD2(I))
     &  CALL CCPERR(1,'ERROR: Map cell parameter mismatch.')
        J=I+3
        IF (ABS(CCD2(J)-CCD1(J)).GT.1E-4*CCD2(J))
     &  CALL CCPERR(1,'ERROR: Map cell parameter mismatch.')
        IF (IP2(I).NE.IP1(I) .OR. LUVW2(I).NE.LUVW1(I) .OR.
     &  MUVW2(I).NE.MUVW1(I) .OR. NXYZ2(I).NE.NXYZ1(I))
     &  CALL CCPERR(1,'ERROR: Map format mismatch.')
        NC=NC*NXYZ1(I)
        VC=VC*CCD1(I)
        IF (CCD1(I+3).EQ.90.) THEN
          T=0.
        ELSE
          C=COS(DR*CCD1(I+3))
          S=S-C**2
          T=T*C
        ENDIF
      ENDDO
      VC=VC*SQRT(S+T)/NC
      WRITE(6,'(/A,F9.4)') 'VC(in) =',VC
C
C#### Get orthogonalisation matrix.
      CALL ORTHOG(CCD1,OMCD)
C
C#### Convert matrix to grid co-ords & get cos & sin of rotation angles.
      DO I=1,3
        DO J=1,3
          K=IP1(J)
          OMGD(I,J)=OMCD(I,K)/NXYZ1(K)
        ENDDO
        CA(I)=COS(DR*AV(I))
        SA(I)=SIN(DR*AV(I))
      ENDDO
C
C#### Get orthogonal rotation matrix.
      RMO(1,1)=CA(1)*CA(2)*CA(3)-SA(1)*SA(3)
      RMO(1,2)=-CA(1)*CA(2)*SA(3)-SA(1)*CA(3)
      RMO(1,3)=CA(1)*SA(2)
      RMO(2,1)=SA(1)*CA(2)*CA(3)+CA(1)*SA(3)
      RMO(2,2)=-SA(1)*CA(2)*SA(3)+CA(1)*CA(3)
      RMO(2,3)=SA(1)*SA(2)
      RMO(3,1)=-SA(2)*CA(3)
      RMO(3,2)=SA(2)*SA(3)
      RMO(3,3)=CA(2)
C
C#### Get rotation matrix & translation vector for input grid co-ords.
      DO I=1,3
        CCD(I)=CCG(I)
        CCG(I)=CCG(I)/NXYZ(I)
        DO J=1,3
          RMG1(I,J)=0.
          DO K=1,3
            RMG1(I,J)=RMG1(I,J)+RMO(I,K)*OMGD(K,J)
          ENDDO
          RMG1(I,J)=RMG1(I,J)/CCG(I)
        ENDDO
        TVG1(I)=TVO(I)/CCG(I)
      ENDDO
C
C#### Allocate memory for input maps & read in maps.
      TALC(1)='B'
      LALC(1)=(MU-LU+1)*(MV-LV+1)
      TALC(2)='R'
      LALC(2)=LALC(1)
      CALL CCPALC(GETDEN,2,TALC,LALC)
C
C#### Get output cell volume & rotation matrix & translation vector for
C#### output grid co-ords.
      NC=1
      VC=1.
      CALL MINV3(RMG1,RMG2,T)
      DO I=1,3
        NC=NC*NXYZ(I)
        VC=VC*CCD(I)
        TVG2(I)=0.
        DO J=1,3
          TVG2(I)=TVG2(I)-RMG2(I,J)*TVG1(J)
        ENDDO
      ENDDO
      VC=VC/NC
      WRITE(6,'(/A,F9.4)') 'VC(out) =',VC
C
C#### Get root of output filenames, open output map & add P1 symmetry.
      CALL GETARG(2,FN)
      L=LENSTR(FN)
      A='Produced by rotmap.'
      CALL MWRHDL(3,FN(:L)//'.map',A,NXYZ(2),IP,NXYZ,0,0,NXYZ(3)-1,0,
     &NXYZ(1)-1,CCD,1,2)
      CALL MSYPUT(2,1,3)
C
C#### Allocate memory for output map & write out map.
      TALC(3)='R'
      LALC(3)=LALC(1)*(MW-LW+1)
      LALC(2)=(LALC(3)+1)/2
      TALC(4)='R'
      LALC(4)=NXYZ(3)*NXYZ(1)
      CALL CCPALC(PUTDEN,4,TALC,LALC)
      CALL MWCLOSE(3)
C
      CALL CCPERR(0,'NORMAL TERMINATION')
1     CALL CCPERR(1,'ERROR: Solvent density format.')
      END
C
C#########################################################################
C
      SUBROUTINE MAPHEAD(IUN,LN,T,IP,LUVW,MUVW,NXYZ,NSG,MM,CCD,RHOA,
     &RHOS)
C#### Read map header.
C
      IMPLICIT NONE
C
      CHARACTER LN*(*), T*80
      INTEGER IUN, MM, NSG, NW
      REAL RHOA, RHOL, RHOS, RHOU
C
      INTEGER IP(3), LUVW(3), MUVW(3), NXYZ(3)
      REAL CCD(6)
C
      CALL MRDHDR(IUN,LN,T,NW,IP,NXYZ,LUVW(3),LUVW(1),MUVW(1),LUVW(2),
     &MUVW(2),CCD,NSG,MM,RHOL,RHOU,RHOA,RHOS)
      MUVW(3)=LUVW(3)+NW-1
      END
C
C#########################################################################
C
```

```
      SUBROUTINE ORTHOG(CCD,OMCD)
C#### Orthogonalisation matrix.
C
      IMPLICIT NONE
C
      INTEGER I
      REAL CAR,DR,PI
      PARAMETER(PI=3.14159265,DR=PI/180.)
C
      REAL CAD(3),CCD(6),OMCD(3,3),SAD(3)
C
C#### Cosines & sines of direct cell angles.
      DO I=1,3
        IF (CCD(3+I).EQ.90.) THEN
          CAD(I)=0.
          SAD(I)=1.
        ELSE
          CAD(I)=COS(DR*CCD(3+I))
          SAD(I)=SIN(DR*CCD(3+I))
        ENDIF
      ENDDO
C
C#### Cos(alpha*).
      CAR=(CAD(2)*CAD(3)-CAD(1))/(SAD(2)*SAD(3))
C
C#### Direct space orthogonalisation matrix.
      OMCD(1,1)=CCD(1)
      OMCD(1,2)=CCD(2)*CAD(3)
      OMCD(1,3)=CCD(3)*CAD(2)
      OMCD(2,1)=0.
      OMCD(2,2)=CCD(2)*SAD(3)
      OMCD(2,3)=-CCD(3)*CAR*SAD(2)
      OMCD(3,1)=0.
      OMCD(3,2)=0.
      OMCD(3,3)=CCD(3)*SQRT(1.-CAR**2)*SAD(2)
C
C      WRITE(6,'()')
C      WRITE(6,*) 'CCD:',CCD
C      WRITE(6,*) 'CAD:',CAD
C      WRITE(6,*) 'SAD:',SAD
      END
C
C#########################################################################
C
      SUBROUTINE MINV3(A,B,D)
C#### Invert 3x3 matrix.
C
      IMPLICIT NONE
      INTEGER I1,I2,I3,J1,J2,J3
      REAL D
C
      INTEGER IP(3)
      REAL A(3,3),B(3,3)
      DATA IP /2,3,1/
C
      D=0.
      DO I1=1,3
        I2=IP(I1)
        I3=IP(I2)
        D=D+A(1,I1)*(A(2,I2)*A(3,I3)-A(2,I3)*A(3,I2))
      ENDDO
```

```
C
      IF (D.NE.0.) THEN
        D=1./D
        DO I1=1,3
          I2=IP(I1)
          I3=IP(I2)
          DO J1=1,3
            J2=IP(J1)
            J3=IP(J2)
            B(J1,I1)=D*(A(I2,J2)*A(I3,J3)-A(I2,J3)*A(I3,J2))
          ENDDO
        ENDDO
      ENDIF
      END
C
C##########################################################################
C
      SUBROUTINE GETDEN(L1,BRHO,L2,RHO)
C#### Read in mask & map data.
C
      IMPLICIT NONE
C
      INTEGER I,IU,IV,IW,J,L1,L2,LU,LV,LW,MU,MV,MW,NC,NM,NMOL,NS
      REAL RHO0,SM,VC
C
      LOGICAL*1 BRHO(LU:MU,LV:MV)
      INTEGER IUVW(3),LXYZ(3),MXYZ(3),NXYZ(3)
      REAL AV(3),CCG(3),RMG1(3,3),RMG2(3,3),RHO(LU:MU,LV:MV),TVG1(3),
     &TVG2(3),TVO(3),XYZ(3)
      COMMON/INPCOM/ CCG,NXYZ,NMOL,AV,TVO
      COMMON/MAPCOM/ LU,LV,LW,MU,MV,MW,NC,RHO0,VC,RMG1,RMG2,TVG1,TVG2
      EQUIVALENCE(IU,IUVW(1)),(IV,IUVW(2)),(IW,IUVW(3))
C
C#### Initialise density sums & grid limits.
      NM=0
      SM=0.
      DO I=1,3
        LXYZ(I)=32767
        MXYZ(I)=-32767
      ENDDO
C
C#### Loop over sections & read into section arrays.
      DO IW=LW,MW
        CALL MGULP0(1,BRHO(LU,LV),I)
        IF (I.NE.0) CALL CCPERR(1,'ERROR in MGULP0.')
        CALL MGULP2(2,RHO(LU,LV),I)
        IF (I.NE.0) CALL CCPERR(1,'ERROR in MGULP2.')
C
C#### For all grid points within the mask, sum density & get grid
C#### limits.
        DO IV=LV,MV
          DO IU=LU,MU
            IF (BRHO(IU,IV)) THEN
              NM=NM+1
              SM=SM+RHO(IU,IV)
              DO I=1,3
                XYZ(I)=TVG1(I)
                DO J=1,3
                  XYZ(I)=XYZ(I)+RMG1(I,J)*IUVW(J)
                ENDDO
                LXYZ(I)=MIN(LXYZ(I),NINT(XYZ(I)))
                MXYZ(I)=MAX(MXYZ(I),NINT(XYZ(I)))
              ENDDO
            ENDIF
          ENDDO
        ENDDO
      ENDDO
C
C#### Check limits fall within output map.
      WRITE(6,'(3(/A,3I5))') 'LXYZ:',LXYZ,'MXYZ:',MXYZ,'NXYZ:',NXYZ
      DO I=1,3
        IF (LXYZ(I).LE.0 .OR. MXYZ(I).GE.NXYZ(I)) THEN
C          WRITE(6,'(3(/A,3I5))') 'LXYZ:',LXYZ,'MXYZ:',MXYZ,'NXYZ:',
C     &    (MXYZ(I)-LXYZ(I)+1,I=1,3)
          CALL CCPERR(1,'ERROR: Increase output CELL & GRID.')
        ENDIF
      ENDDO
C
C#### Compute various properties of density.
      IF (NM.EQ.0) CALL CCPERR(1,'ERROR: Protein volume = 0.')
      RHO0=SM/NM
      WRITE(6,'(/A/A,I9,F9.4)') 'For input map:',
     &'Protein volume, mean density =',NINT(NM*VC),RHO0
      NS=NC-NMOL*NM
      IF (NS.LE.0) CALL CCPERR(1,'ERROR: Solvent volume <= 0.')
      WRITE(6,'(A,I9,F9.4)')
     &'Solvent volume, mean density =',NINT(NS*VC/NMOL),-NMOL*SM/NS
      WRITE(6,'(A,9X,F8.3/)') 'Solvent volume fraction      =',
     &REAL(NS)/NC
      END
C
C#######################################################################
C
      SUBROUTINE PUTDEN(L1,BRHOR,L2,IRHOR,L3,RHOR,L4,RHOW)
C#### Re-read input maps & write out rotated & interpolated map.
C
      IMPLICIT NONE
C
      LOGICAL LB,LM
      INTEGER I,IU,IU0,IU1,IV,IV0,IV1,IW,IW0,IW1,IX,IY,IZ,J,JU,JV,JW,L1,
     &L2,L3,L4,LU,LV,LW,MU,MV,MW,NC,NM,NMOL,NS,NX,NY,NZ
      REAL R0,R1,RHO0,SM,VC
C
      LOGICAL*1 BRHOR(LU:MU,LV:MV)
      INTEGER*2 IRHOR(LU:MU,LV:MV,LW:MW)
      INTEGER IUVW0(3),IUVW1(3),IXYZ(3),LUVW(3),LUVW1(3),MUVW(3),
     &MUVW1(3),NXYZ(3)
      REAL AV(3),CCG(3),RMG1(3,3),RMG2(3,3),RHOR(LU:MU,LV:MV,LW:MW),
     &RHOW(0:NZ-1,0:NX-1),TVG1(3),TVG2(3),TVO(3),UVW(3),UVWA(3)
      COMMON/INPCOM/ CCG,NXYZ,NMOL,AV,TVO
      COMMON/MAPCOM/ LU,LV,LW,MU,MV,MW,NC,RHO0,VC,RMG1,RMG2,TVG1,TVG2
      REAL QINT3D
      EQUIVALENCE
     &(IU0,IUVW0(1)),(IV0,IUVW0(2)),(IW0,IUVW0(3)),
     &(IU1,IUVW1(1)),(IV1,IUVW1(2)),(IW1,IUVW1(3)),
     &(IX,IXYZ(1)),(IY,IXYZ(2)),(IZ,IXYZ(3)),
     &(LU,LUVW1(1)),(MU,MUVW1(1)),
     &(NX,NXYZ(1)),(NY,NXYZ(2)),(NZ,NXYZ(3))
C
C#### Rewind input maps and read in again.
C      WRITE(6,'(2(/A,3I5))') 'LUVW1:',LUVW1,'MUVW1:',MUVW1
      CALL MPOSN(1,LW)
      CALL MPOSN(2,LW)
      DO IW=LW,MW
        CALL MGULP0(1,BRHOR(LU,LV),I)
        IF (I.NE.0) CALL CCPERR(1,'ERROR in MGULP0.')
        CALL MGULP2(2,RHOR(LU,LV,IW),I)
        IF (I.NE.0) CALL CCPERR(1,'ERROR in MGULP2.')
C
C#### Set flag for each grid point inside mask.
        DO IV=LV,MV
          DO IU=LU,MU
            IF (.NOT.BRHOR(IU,IV)) THEN
              IRHOR(IU,IV,IW)=0
            ELSE
              IRHOR(IU,IV,IW)=1
            ENDIF
          ENDDO
        ENDDO
      ENDDO
C
C#### Initialise limits for output map.
      DO I=1,3
        LUVW(I)=MUVW1(I)
        MUVW(I)=LUVW1(I)
      ENDDO
C
C#### For each grid point in output map find the corresponding input
C#### point & set flag if it falls outside input limits.
C      WRITE(6,'()')
      NM=0
      SM=0.
      DO IY=0,NY-1
        DO IX=0,NX-1
          DO IZ=0,NZ-1
            LB=.FALSE.
            DO I=1,3
              UVW(I)=TVG2(I)
              DO J=1,3
                UVW(I)=UVW(I)+RMG2(I,J)*IXYZ(J)
              ENDDO
              IUVW0(I)=NINT(UVW(I))
              IF (IUVW0(I).LE.LUVW1(I) .OR. IUVW0(I).GE.MUVW1(I))
     &        LB=.TRUE.
            ENDDO
C
C#### Set flag if any of the 27 points in 3x3x3 box centred on nearest
C#### grid point are inside the mask.
            LM=.FALSE.
            DO JW=-1,1
              IW=IW0+JW
              IF (IW.GE.LUVW1(3) .AND. IW.LE.MUVW1(3)) THEN
                DO JV=-1,1
                  IV=IV0+JV
                  IF (IV.GE.LUVW1(2) .AND. IV.LE.MUVW1(2)) THEN
                    DO JU=-1,1
                      IU=IU0+JU
                      IF (IU.GE.LUVW1(1) .AND. IU.LE.MUVW1(1)) THEN
                        IF (IRHOR(IU,IV,IW).GT.0) LM=.TRUE.
                      ENDIF
                    ENDDO
                  ENDIF
                ENDDO
              ENDIF
            ENDDO
C
C#### If so update limits for output map.
            IF (LM) THEN
              DO I=1,3
                LUVW(I)=MIN(LUVW(I),IUVW0(I))
                MUVW(I)=MAX(MUVW(I),IUVW0(I))
              ENDDO
            ENDIF
C
C#### Set output density to zero if point outside input map.
            IF (LB) THEN
              RHOW(IZ,IX)=0.
C
C#### Otherwise get fractional grid co-ords for interpolation.
            ELSE
              DO I=1,3
                UVW(I)=UVW(I)-IUVW0(I)
                UVWA(I)=ABS(UVW(I))
                IUVW1(I)=IUVW0(I)+INT(SIGN(1.,UVW(I)))
              ENDDO
C
C#### Interpolate the mask.
              R0=IRHOR(IU0,IV0,IW0)+UVWA(1)*(IRHOR(IU1,IV0,IW0)-
     &        IRHOR(IU0,IV0,IW0))
              R1=IRHOR(IU0,IV0,IW1)+UVWA(1)*(IRHOR(IU1,IV0,IW1)-
     &        IRHOR(IU0,IV0,IW1))
              R0=R0+UVWA(2)*(IRHOR(IU0,IV1,IW0)+UVWA(1)*
     &        (IRHOR(IU1,IV1,IW0)-IRHOR(IU0,IV1,IW0))-R0)
C
C#### If interpolated mask value < 0.5225, set output density to zero.
C#### This magic figure seems to give the right mask volume!
              IF (R0+UVWA(3)*(R1+UVWA(2)*(IRHOR(IU0,IV1,IW1)+UVWA(1)*
     &        (IRHOR(IU1,IV1,IW1)-IRHOR(IU0,IV1,IW1))-R1)-R0).LT..5225)
     &        THEN
                RHOW(IZ,IX)=0.
C
C#### Otherwise interpolate input density & sum output density.
              ELSE
                RHOW(IZ,IX)=QINT3D(LU,MU,LV,MV,LW,MW,RHOR,IUVW0,UVW)-
     &          RHO0
                NM=NM+1
                SM=SM+RHOW(IZ,IX)
              ENDIF
            ENDIF
          ENDDO
        ENDDO
C
C#### Write out zx map section.
        CALL MSPEW(3,RHOW)
      ENDDO
C
      WRITE(6,'(4(/A,3I5))') 'LUVW1:',LUVW1,'LUVW :',LUVW,'MUVW :',
     &MUVW,'MUVW1:',MUVW1
C
C#### Check that all required points were in input map.
      DO I=1,3
        IF (LUVW(I).LE.LUVW1(I) .OR. MUVW(I).GE.MUVW1(I)) THEN
C          WRITE(6,'(4(/A,3I5))') 'LUVW1:',LUVW1,'LUVW :',LUVW,'MUVW :',
C    &     MUVW,'MUVW1:',MUVW1
```

```
          CALL CCPERR(1,'ERROR: Required grid point not in map.')
        ENDIF
      ENDDO
C
      IF (NM.EQ.0) CALL CCPERR(1,'ERROR: Protein volume = 0.')
      IF (NM.EQ.NC) CALL CCPERR(1,'ERROR: Solvent volume = 0.')
C
C#### Write out some properties of output map.
      WRITE(6,'(/A/A,I9,F9.4)') 'For output map:',
     &'Protein volume, mean density    =',NINT(NM*VC),SM/NM
      NS=NC-NM
      WRITE(6,'(A,I9,F9.4)') 'Solvent volume, volume fraction =',
     &NINT(NS*VC),REAL(NS)/NC
      END
C
C#########################################################################
C
      REAL FUNCTION QINT3D(LX,MX,LY,MY,LZ,MZ,R,IXYZ,XYZ)
C#### 27-point quadratic interpolation on a 3-D cubic grid.
C
      IMPLICIT NONE
      INTEGER I,IX,IY,IZ,J,JX,JY,JZ,K,LX,LY,LZ,MX,MY,MZ
      INTEGER II(3),IXYZ(3),JXYZ(3)
      REAL BV(10),DV(10),R(LX:MX,LY:MY,LZ:MZ),XYZ(3)
      EQUIVALENCE(JX,JXYZ(1)),(JY,JXYZ(2)),(JZ,JXYZ(3))
      DATA II/2,3,1/
C
      IF (IXYZ(1).LE.LX .OR. IXYZ(1).GE.MX .OR. IXYZ(2).LE.LY .OR.
     &IXYZ(2).GE.MY .OR. IXYZ(3).LE.LZ .OR. IXYZ(3).GE.MZ) THEN
        WRITE(6,'(3(/A,3I5)') 'LXYZ :',LX,LY,LZ,'MXYZ :',MX,MY,MZ,
     &  'IXYZ:',IXYZ
        CALL CCPERR(1,'ERROR in QINT3D.')
      ENDIF
C
C#### Initialise least-squares coeffients.
      DO I=1,10
        BV(I)=0.
      ENDDO
C
C#### Get derivative vector & accumulate RHS for least-squares.
      DV(1)=1.
      DO JZ=-1,1
        IZ=IXYZ(3)+JZ
        DO JY=-1,1
          IY=IXYZ(2)+JY
          DO JX=-1,1
            IX=IXYZ(1)+JX
            DO I=1,3
              J=II(I)
              K=II(J)
              DV(1+I)=JXYZ(I)
              DV(4+I)=JXYZ(I)**2
              DV(7+I)=JXYZ(J)*JXYZ(K)
            ENDDO
            DO I=1,10
              BV(I)=BV(I)+DV(I)*R(IX,IY,IZ)
            ENDDO
          ENDDO
        ENDDO
      ENDDO
C
```

```
C#### Solve the equations for the coefficients & compute the
C#### interpolated density.
      QINT3D=7.*BV(1)/27.-(BV(5)+BV(6)+BV(7))/9.
      DO I=2,4
        DV(I)=BV(I)/18.
      ENDDO
      BV(1)=BV(1)/9.
      DO I=5,7
        DV(I)=BV(I)/6.-BV(1)
      ENDDO
      DO I=8,10
        DV(I)=BV(I)/12.
      ENDDO
C      WRITE(6,'(/10F8.4/)') DV
C
      DO I=1,3
        J=II(I)
        K=II(J)
        QINT3D=QINT3D+(DV(1+I)+DV(4+I)*XYZ(I))*XYZ(I)+
     &  DV(7+I)*XYZ(J)*XYZ(K)
      ENDDO
      END
C
C#########################################################################
C
      INCLUDE 'namelist.f'
```

ANNEX 2

```
      PROGRAM DENCOR
C
C#### Compute density correlation coefficient for 2 maps (CCP4 format).
C#### Memory for map storage is allocated dynamically.
C
C#### Ian J. Tickle, Astex Technology.
C#### Copyright © 2003 Astex Technology Ltd.  All rights reserved.
C
C#### Link with standard CCP4 libraries (no additional routines
C#### required).
C
C#### Usage:
C
C#### Command-line input:
C#### dencor  MAPIN1 filename_map1  MAPIN2 filename_map2
C
C#### Standard input:
C#### None.
C
C#### Output:
C#### Last line of standard output is the correlation coefficient,
C#### expressed as a percentage.
C
      IMPLICIT NONE
      CHARACTER A*80
      INTEGER I,J,LU,LV,LW,MM,MU,MV,MW,NS1,NS2
      REAL RHOA,RHOS
C
      CHARACTER TALC(2)
      INTEGER IP1(3),IP2(3),LALC(2),LUVW1(3),LUVW2(3),MUVW1(3),MUVW2(3),
     &NXYZ1(3),NXYZ2(3)
      REAL CCD1(6),CCD2(6)
C
      COMMON/MAPCOM/ LU,LV,LW,MU,MV,MW
      EQUIVALENCE(LU,LUVW1(1)),(MU,MUVW1(1))
      EXTERNAL GETDEN
      DATA TALC/2*'R'/
C
C#### Read & check CCP4 map headers for consistency.
      CALL CCPFYP
      CALL MAPHEAD(1,'MAPIN1',A,IP1,LUVW1,MUVW1,NXYZ1,NS1,MM,CCD1,RHOA,
     &RHOS)
      IF (MM.NE.2) CALL CCPERR(1,'Map mode must be 2.')
      CALL MAPHEAD(2,'MAPIN2',A,IP2,LUVW2,MUVW2,NXYZ2,NS2,MM,CCD2,RHOA,
     &RHOS)
      IF (MM.NE.2) CALL CCPERR(1,'Map mode must be 2.')
      IF (NS1.NE.NS2) CALL CCPERR(1,'Space group mismatch.')
C
      DO I=1,3
        IF (ABS(CCD2(I)-CCD1(I)).GT.1E-5*CCD2(I))
     &  CALL CCPERR(1,'ERROR: Map cell parameter mismatch.')
        J=I+3
        IF (ABS(CCD2(J)-CCD1(J)).GT.1E-4*CCD2(J))
     &  CALL CCPERR(1,'ERROR: Map cell parameter mismatch.')
        IF (IP2(I).NE.IP1(I) .OR. LUVW2(I).NE.LUVW1(I) .OR.
     &  MUVW2(I).NE.MUVW1(I) .OR. NXYZ2(I).NE.NXYZ1(I))
     &  CALL CCPERR(1,'ERROR: Map format mismatch.')
```

```
      ENDDO
C
C#### Allocate memory to store maps.
      LALC(1)=(MU-LU+1)*(MV-LV+1)
      LALC(2)=LALC(1)
      CALL CCPALC(GETDEN,2,TALC,LALC)
      END
C
C#######################################################################
C
      SUBROUTINE MAPHEAD(IUN,LN,T,IP,LUVW,MUVW,NXYZ,NSG,MM,CCD,RHOA,
     &RHOS)
C#### Read map header.
C
      IMPLICIT NONE
C
      CHARACTER LN*(*), T*80
      INTEGER I, IUN, MM, NSG, NW
      REAL RHOA, RHOL, RHOS, RHOU
C
      INTEGER IP(3), LUVW(3), MUVW(3), NXYZ(3)
      REAL CCD(6)
C
      I=0
      CALL MRDHDS(IUN,LN,T,NW,IP,NXYZ,LUVW(3),LUVW(1),MUVW(1),LUVW(2),
     &MUVW(2),CCD,NSG,MM,RHOL,RHOU,RHOA,RHOS,I,0)
      MUVW(3)=LUVW(3)+NW-1
      END
C
C#######################################################################
C
      SUBROUTINE GETDEN(L1,RHO1,L2,RHO2)
C#### Read in map densities and compute density correlation coefficient.
      IMPLICIT NONE
C
      INTEGER I,IU,IV,IW,L1,L2,LU,LV,LW,MU,MV,MW
      REAL S11,S12,S22
C
      REAL RHO1(LU:MU,LV:MV),RHO2(LU:MU,LV:MV)
      COMMON/MAPCOM/ LU,LV,LW,MU,MV,MW
C
C#### Initialise sums for correlation coefficient.
      S11=0.
      S22=0.
      S12=0.
C
C#### Loop over map sections.
      DO IW=LW,MW
C
C#### Read in section for each map.
        CALL MGULP(1,RHO1,I)
        IF (I.NE.0) CALL CCPERR(1,'ERROR in MGULP.')
        CALL MGULP(2,RHO2,I)
        IF (I.NE.0) CALL CCPERR(1,'ERROR in MGULP.')
C
C#### Accumulate sums for correlation coefficient over section.
        DO IV=LV,MV
          DO IU=LU,MU
            S11=S11+RHO1(IU,IV)**2
            S22=S22+RHO2(IU,IV)**2
            S12=S12+RHO1(IU,IV)*RHO2(IU,IV)
          ENDDO
        ENDDO
      ENDDO
C
C#### Write out correlation coefficient.
      WRITE(6,'(I3)') NINT(100.*S12/SQRT(S11*S22))
      END
```

ANNEX 3

Astex-EXTENDC

The following modification was made to the source code of the CCP4 program EXTEND (Collaborative Computational Project 4. The CCP4 Suite: Programs for Protein Crystallography, *Acta Crystallographica*, D50, (1994), 760-763.), which is used to extend an electron density map from the asymmetric unit computed by the FFT program (for example to cover a complete protein molecule): the mean and RMS deviation of the electron density are not recalculated for the extended map. Instead in the modified source code these values are simply copied over from the original map computed by the FFT program.

The rationale for this is statistical rigour: the asymmetric unit of the map represents the true population of electron density values in the statistical sense and therefore the values of the mean and RMS deviation for the asymmetric unit can be considered to be those of the true population mean and population standard deviation respectively. Any other subset or superset of the map that is not an integral number of asymmetric units is a sample in the statistical sense, with corresponding sample mean and sample standard deviation. It may even be a biased sample because when a map is extended, density values related by the map symmetry are very likely to be generated more than once. Sample statistics, whether biased or not, are always only approximations to population statistics.

ANNEX 4

```
      PROGRAM KFIT
C
C#### Sequence/structure-based alignment & fitting using Needleman/
C#### Wunsch/Sellers & Kearsley's algorithms respectively:
C#### Needleman, S.B. & Wunsch, C.D., J. Mol. Biol. (1970) 48: 443-453.
C#### Sellers, P.H. J. Appl. Math. (1974) 26: 787-793.
C#### Kearsley, S.K., Acta Cryst. (1989) A45: 208-210.
C#### Memory for scoring matrix is allocated dynamically.
C
C#### Ian J. Tickle, Astex Technology.
C#### Copyright © 2002-2003 Astex Technology Ltd.  All rights reserved.
C
C#### Link with standard CCP4 libraries.
C#### Additional routine required: ldigr.f .
C
C#### Usage:
C
C#### Command-line input:
C#### kfit
C
C#### Standard input:
C#### KEYWORD VALUE pairs, newline-separated.
C
C#### Keywords & associated values:
C
C#### STRU p      PDB filename (first given is treated as static
C####             molecule, all subsequent given are fitted).
C#### MOLE c      Concatenated list of chain id(s) which define a
C####              molecule.
C#### RESI r1 r2  Residue number range for fitting.  Default = all.
C#### SMAX s      Max sequence identity criterion.  Default = 1.
C#### DMAX d      Max distance criterion for selection of atom pairs for
C####             fitting.  Default = 1, give large value (e.g. 999) for
C####             automatic selection.
C#### BCOL b      B-factor mode:
C####             KEEP  keep original B factors.
C####             RMSD  replace B factor with distance deviation.
C####             SET x replace B factor with value x.
C#### FIT  f      Optional filename for fitted structure.
C#### TRAN t      Filename to transform with same matrix.
C
C#### Unix example:
C
C#### kfit <<EOD
C#### stru 2C5-apo-p1box.pdb
C#### stru 3A4-apo.pdb
C#### dmax 999
C#### fit /tmp/ian/3A4-apo-rotmap1-kfit.pdb
C#### EOD
C
C
      IMPLICIT NONE
      CHARACTER A*255,B*255,RP*6,RS*8
      LOGICAL LF,LR1,LR2
      INTEGER I,I1,I2,IA,IA1,IA2,IAF,IBC,IN,IR,IR1,IR2,IRF,IRP,IS,ISF,
     &IT,IW,J,JA1,JA2,JT,JW,K,L,MAP,MIT,MRP,N,NA,NAF,NAI,NAT,NB,NCM,NS
      REAL BF,CA3,CA31,D,DM,DMAX,DMAXS,PI,RD,S,SA3,SMAX
      REAL*8 D1,D2
      PARAMETER(MRP=4000,MAP=10*MRP,MIT=100,PI=3.141592654,RD=180./PI)
C
      CHARACTER AA(3)*2,ANP(MAP,2)*14,BA(2)*255,BS(2)*255,CSF(2)*9,
     &RSF(2,MRP,2)*8,RNP(MRP,2)*8,RTP(MRP,2)*3
      LOGICAL LA(3),LRP(2,MRP),LSF(MRP,2)
      INTEGER INRP(MAP+1,2),IRAP(0:MRP,2),IV(8),JV(40),JRP(MRP+1),
     &KAF(MAP,2),KAP(MAP,2),KRF(MRP,2),NAFS(MIT),NAP(2),NCP(MAP),NRF(2),
     &NRP(2),NSF(2)
      REAL AV(3),DAP(MAP),TV(3),V(3),X1(3),X2(3),XAP(3,MAP,3),XC(3,2),
     &XCS(3,2,MIT),XL(3),XM(3)
      REAL*8 AM(4,4),EV(4),RM(4,4),WV(132),XD(3),XS(3)
C
      INTEGER LDIGA,LDIGF,LDIGR,LENSTR
C
      DATA AA/'N','CA','C'/
C
C#### Set defaults.
      CALL CCPFYP
      SMAX=1.
      DMAX=1.
      IBC=0
      NS=0
      NRF(1)=0
      NRF(2)=0
C
C#### Read standard input.
1     IF (ISATTY(6)) WRITE(6,'(/A,$)') 'Input: '
      IF (LDIGR(5,A).LT.0) GOTO 6
      IF (LDIGA(1,B).EQ.0) GOTO 1
      IF (.NOT.ISATTY(6)) WRITE(6,'(/2A)') 'Input line: ',A(:LENSTR(A))
      CALL CCPUPC(B)
C
      L=LENSTR(B)
      IF (L.GE.4 .AND. B.EQ.'STRUCTURE'(:L)) THEN
        IF (NS.LT.2) THEN
          NS=NS+1
        ELSE
          CLOSE(1)
        ENDIF
        IF (LDIGA(1,BS(NS)).EQ.0) THEN
          CALL ERRMSG('ERROR: Filename missing.')
          NS=NS-1
          GOTO 1
        ENDIF
        INQUIRE(FILE=BS(NS),EXIST=LF)
        IF (.NOT.LF) THEN
          CALL ERRMSG('ERROR: Non-existent file.')
          NS=NS-1
          GOTO 1
        ENDIF
        CSF(NS)='  '
        NSF(NS)=0
        NRP(NS)=0
        NAP(NS)=0
C
      ELSEIF (L.GE.4 .AND. B.EQ.'MOLECULE'(:L)) THEN
        IF (NS.EQ.0) THEN
          CALL ERRMSG('ERROR: No files read.')
          GOTO 1
        ENDIF
        IF (CSF(NS).NE.' ') THEN
          CALL ERRMSG('ERROR: Duplicate molecule specified.')
          GOTO 1
        ENDIF
        IF (LDIGA(1,CSF(NS)).EQ.0) THEN
          CALL ERRMSG('ERROR: No chain id(s).')
          GOTO 1
        ENDIF
C        WRITE(6,'(/3A)') '=',CSF(NS),'='
C
      ELSEIF (L.GE.4 .AND. B.EQ.'RESIDUE'(:L)) THEN
        IF (NS.EQ.0) THEN
          CALL ERRMSG('ERROR: No files read.')
          GOTO 1
        ENDIF
        IF (NSF(NS).EQ.MRP) CALL CCPERR(1,'ERROR: Increase MRP.')
        NSF(NS)=NSF(NS)+1
        RSF(1,NSF(NS),NS)=''
        N=LDIGA(2,RSF(1,NSF(NS),NS))
        IF (RSF(1,NSF(NS),NS).EQ.'') THEN
          CALL ERRMSG('ERROR: No residue number(s).')
          GOTO 1
        ENDIF
        LSF(NSF(NS),NS)=INDEX(RSF(1,NSF(NS),NS),':').GT.0
        IF (N.EQ.1) THEN
          RSF(2,NSF(NS),NS)=RSF(1,NSF(NS),NS)
        ELSE
          IF (RSF(1,NSF(NS),NS).EQ.'*') THEN
            LSF(NSF(NS),NS)=INDEX(RSF(2,NSF(NS),NS),':').GT.0
          ELSEIF (RSF(2,NSF(NS),NS).NE.'*' .AND.
     &    LSF(NSF(NS),NS).NEQV.INDEX(RSF(2,NSF(NS),NS),':').GT.0) THEN
            CALL ERRMSG('ERROR: Invalid residue range.')
            GOTO 1
          ENDIF
        ENDIF
        IF (NSF(NS).GT.1) THEN
          IF (RSF(1,NSF(NS),NS).EQ.'*' .OR. RSF(2,NSF(NS)-1,NS).EQ.'*')
     &    THEN
            CALL ERRMSG('ERROR: Invalid residue range.')
            GOTO 1
          ENDIF
        ENDIF
C
      ELSEIF (B.EQ.'SMAX') THEN
        IF (LDIGF(1,SMAX).EQ.0) THEN
          CALL ERRMSG('ERROR: Missing value.')
          GOTO 1
        ENDIF
C
      ELSEIF (B.EQ.'DMAX') THEN
        IF (LDIGF(1,DMAX).EQ.0) THEN
          CALL ERRMSG('ERROR: Missing value.')
          GOTO 1
        ENDIF
        DMAX=MAX(DMAX,0.)
C
      ELSEIF (B.EQ.'BCOL') THEN
        IF (LDIGA(1,B).EQ.0) THEN
          CALL ERRMSG('ERROR: Missing value.')
          GOTO 1
        ENDIF
        CALL CCPUPC(B)
        IF (B.EQ.'KEEP') THEN
          IBC=0
        ELSEIF (B.EQ.'RMSD') THEN
          IBC=-1
        ELSEIF (B.EQ.'SET') THEN
          BF=0.
          IF (LDIGF(1,BF).EQ.0) THEN
            CALL ERRMSG('ERROR: Missing value.')
            GOTO 1
          ENDIF
          IBC=1
        ELSE
          CALL ERRMSG('ERROR: Bad keyword.')
        ENDIF
C
C#### FIT keyword drives fitting, first check enough files read in.
      ELSEIF (B.EQ.'FIT') THEN
        IF (NS.LT.2) THEN
          CALL ERRMSG('ERROR: Need 2 or more files.')
          GOTO 1
        ENDIF
C
C#### Read in PDB files and store atomic data.
        DO IS=1,2
          IF (NAP(IS).EQ.0) THEN
            OPEN(1,FILE=BS(IS),STATUS='OLD')
            RP=CHAR(0)
2           READ(1,'(A)',END=3) A
            IF (A(:6).EQ.'ATOM' .AND. (IS.EQ.1 .OR. A(13:13).EQ.' '
     &      .AND. (A(14:14).EQ.'C' .AND. INDEX('AB',A(15:15)).GT.0 .OR.
     &      A(15:15).EQ.' ') .AND. A(16:16).EQ.' ') .AND.
     &      A(18:20).NE.'HOH' .AND. A(18:20).NE.'WAT' .AND.
     &      (CSF(IS).EQ.' ' .OR. INDEX(CSF(IS),A(22:22)).GT.0)) THEN
              IF (A(22:27).NE.RP) THEN
                IF (NRP(IS).EQ.MRP)
     &          CALL CCPERR(1,'ERROR: Increase MRP.')
                RP=A(22:27)
                IF (A(22:22).EQ.' ') THEN
                  L=0
                ELSE
                  L=1
                  RS(:1)=A(22:22)
                ENDIF
                L=L+1
                RS(L:L)=':'
                DO I=23,26
                  IF (A(I:I).NE.' ') THEN
                    J=L+27-I
                    RS(L+1:J)=A(I:26)
                    L=J
                    GOTO 7
                  ENDIF
                ENDDO
7               IF (A(27:27).NE.' ') THEN
                  L=L+2
                  RS(L-1:L)=':'//A(27:27)
                ENDIF
                IF (L.LT.8) RS(L+1:)=' '
                IRAP(NRP(IS),IS)=NAP(IS)
                NRP(IS)=NRP(IS)+1
                RNP(NRP(IS),IS)=RS
                RTP(NRP(IS),IS)=A(18:20)
              ENDIF
              IF (NAP(IS).EQ.MAP) CALL CCPERR(1,'ERROR: Increase MAP.')
              NAP(IS)=NAP(IS)+1
              ANP(NAP(IS),IS)=A(14:27)
              INRP(NAP(IS),IS)=NRP(IS)
              READ(A(31:54),'(3F8.3)') (XAP(I,NAP(IS),IS),I=1,3)
C              WRITE(6,'(I1,2I5,2X,A)') IS,NRP(IS),NAP(IS),A(14:27)
            ENDIF
            GOTO 2
C
3           IF (NAP(IS).EQ.0) THEN
              CALL ERRMSG('ERROR: No atoms!')
              NS=NS-1
              GOTO 1
            ENDIF
            IF (IS.EQ.1) THEN
              CLOSE(1)
            ELSE
              REWIND 1
            ENDIF
            IRAP(NRP(IS),IS)=NAP(IS)
            INRP(NAP(IS)+1,IS)=NRP(IS)+1
          ENDIF
        ENDDO
C
C#### Do the NWS sequence alignment.
C        WRITE(6,'()')
        IF (NSF(1).EQ.0) THEN
          IF (NSF(2).EQ.0) THEN
            CALL SEQALG(SMAX,NRP,RNP,RTP,NRF,KRF)
          ELSE
            NRF(1)=NRP(1)
            DO IR=1,NRF(1)
              KRF(IR,1)=IR
            ENDDO
          ENDIF
        ELSEIF (NSF(2).EQ.0) THEN
          NSF(2)=NSF(1)
          DO ISF=1,NSF(1)
            LSF(ISF,2)=LSF(ISF,1)
            RSF(1,ISF,2)=RSF(1,ISF,1)
            RSF(2,ISF,2)=RSF(2,ISF,1)
          ENDDO
        ENDIF
C
C#### Select the residues for fitting.
        DO IS=1,2
          IF (NSF(IS).GT.0 .AND. (NRF(IS).EQ.0 .OR. IS.EQ.2)) THEN
            WRITE(6,'()')
            ISF=0
            IRP=0
            NRF(IS)=0
21          LR1=.FALSE.
            ISF=ISF+1
22          IRP=IRP+1
            IF (.NOT.LR1) THEN
              IF (LSF(ISF,IS)) THEN
                IF (RSF(1,ISF,IS).EQ.'*' .OR.
     &          RNP(IRP,IS).EQ.RSF(1,ISF,IS)) THEN
```

```
                  LR1=.TRUE.
                  LR2=.FALSE.
                ENDIF
              ELSE
                IF (RSF(1,ISF,IS).EQ.'*' .OR.
     &          RNP(IRP,IS)(:1).EQ.RSF(1,ISF,IS)(:1)) THEN
                  LR1=.TRUE.
                  LR2=.FALSE.
                ENDIF
              ENDIF
            ENDIF
C            WRITE(6,'(I1,I6,2(2X,A),I6,2X,A,2L4)') IS,ISF,
C     &      RSF(1,ISF,IS),RSF(2,ISF,IS),IRP,RNP(IRP,IS),LR1,LR2
            IF (LR1) THEN
              IF (.NOT.LSF(ISF,IS) .AND. LR2 .AND.
     &        RNP(IRP,IS)(:1).NE.RSF(2,ISF,IS)(:1)) THEN
                IF (ISF.LT.NSF(IS)) GOTO 21
                GOTO 23
              ENDIF
              NRF(IS)=NRF(IS)+1
              KRF(NRF(IS),IS)=IRP
              IF (LSF(ISF,IS)) THEN
                IF (RNP(IRP,IS).EQ.RSF(2,ISF,IS)) THEN
                  IF (ISF.LT.NSF(IS)) GOTO 21
                  GOTO 23
                ENDIF
              ELSE
                IF (RNP(IRP,IS)(:1).EQ.RSF(2,ISF,IS)(:1)) LR2=.TRUE.
              ENDIF
            ENDIF
            IF (IRP.LT.NRP(IS)) GOTO 22
            IF (.NOT.LR1) THEN
              CALL ERRMSG('ERROR: Start residue selection not found: '//
     &        RSF(1,ISF,IS)(:LENSTR(RSF(1,ISF,IS))))
              NS=NS-1
              GOTO 1
            ELSEIF (ISF.LT.NSF(IS) .OR. .NOT.LR2 .AND.
     &      RSF(2,ISF,IS).NE.'*') THEN
              CALL ERRMSG('ERROR: End residue selection not found: '//
     &        RSF(2,ISF,IS)(:LENSTR(RSF(2,ISF,IS))))
              NS=NS-1
              GOTO 1
            ENDIF
          ENDIF
23        CONTINUE
        ENDDO
        IF (NRF(2).NE.NRF(1)) THEN
          CALL ERRMSG(
     &    'ERROR: Residue counts in initial alignment differ.')
          NS=NS-1
          GOTO 1
        ENDIF
C
C#### Select the atom pairs for fitting.
C        WRITE(6,'()')
        NAF=0
        DO IRF=1,NRF(1)
          NA=0
          IR1=KRF(IRF,1)
          DO IA1=IRAP(IR1-1,1)+1,IRAP(IR1,1)
            IR2=KRF(IRF,2)
            DO IA2=IRAP(IR2-1,2)+1,IRAP(IR2,2)
              IF (ANP(IA2,2)(:4).EQ.ANP(IA1,1)(:4)) THEN
                NA=NA+1
                NAF=NAF+1
                KAF(NAF,1)=IA1
                KAF(NAF,2)=IA2
C                WRITE(6,'(I6,4X,A,2(4X,A))') NAF,ANP(IA1,1)(:4),
C     &          ANP(IA1,1)(5:),ANP(IA2,2)(5:)
              ENDIF
            ENDDO
          ENDDO
C          IF (NA.EQ.0) CALL CCPERR(1,
C     &    'ERROR: Residues have no common atoms: '//RNP(KRF(IRF,1),1)//
C     &    '  '//RNP(KRF(IRF,2),2))
        ENDDO
        IF (NAF.LE.3) THEN
          CALL ERRMSG('ERROR: <= 3 atoms in initial alignment.')
          NS=NS-1
          GOTO 1
        ENDIF
C
        IF (NSF(1)+NSF(2).GT.0 .AND. NRF(1).GT.3) THEN
          DMAXS=0.
        ELSE
          DMAXS=DMAX**2
        ENDIF
C
        IT=0
        IW=1
C
        DO IA2=1,NAP(2)
          KAP(IA2,2)=0
        ENDDO
C
C#### Determine mean centres of both sets co-ords.
9       DO IS=1,2
          DO I=1,3
            XC(I,IS)=0.
          ENDDO
C
          DO IAF=1,NAF
            IA=KAF(IAF,IS)
C            WRITE(6,'(I1,I6,2X,A,3F8.3)') IS,IA,ANP(IA,IS),
C     &      (XAP(I,IA,IS),I=1,3)
            DO I=1,3
              XC(I,IS)=XC(I,IS)+XAP(I,IA,IS)
            ENDDO
          ENDDO
C
          DO I=1,3
            XC(I,IS)=XC(I,IS)/NAF
          ENDDO
        ENDDO
C
        DO JT=1,IT
          IF (NAF.EQ.NAFS(JT)) THEN
            DO IS=1,2
              DO I=1,3
                IF (XC(I,IS).NE.XCS(I,IS,JT)) GOTO 10
              ENDDO
            ENDDO
            IF (NAF.LE.NAFS(IT)) THEN
              IF (NAF.EQ.NAFS(IT)) IW=3-IW
              GOTO 13
            ENDIF
          ENDIF
10        CONTINUE
        ENDDO
C
C#### Iterate rejection/fitting algorithm.
        IF (DMAXS.GT.0) THEN
          IT=IT+1
          WRITE(6,'(//A,I4/)') 'Iteration',IT
          NAFS(IT)=NAF
          DO IS=1,2
            DO I=1,3
              XCS(I,IS,IT)=XC(I,IS)
            ENDDO
          ENDDO
        ENDIF
C
        WRITE(6,'(A,I2,A,I7,3F9.3)')
     &  ('No of fitting points, centroid for structure',IS,' :',NAF,
     &  (XC(I,IS),I=1,3),IS=1,2)
C
C        WRITE(6,'(/(I5,2(I5,2X,A)))') (IAF,(KAF(IAF,IS),
C     &  ANP(KAF(IAF,IS),IS),IS=1,2),IAF=1,NAF)
C
C#### Kearsley's fitting algorithm using quaternions.
        DO I=1,4
          DO J=1,I
            AM(I,J)=0.
          ENDDO
        ENDDO
C
        DO IAF=1,NAF
          IA1=KAF(IAF,1)
          IA2=KAF(IAF,2)
          DO I=1,3
            D1=XAP(I,IA1,1)-XC(I,1)
            D2=XAP(I,IA2,2)-XC(I,2)
            XD(I)=D1-D2
            XS(I)=D1+D2
          ENDDO
          AM(1,1)=AM(1,1)+XD(1)**2+XD(2)**2+XD(3)**2
          AM(2,1)=AM(2,1)+XS(2)*XD(3)-XD(2)*XS(3)
          AM(2,2)=AM(2,2)+XD(1)**2+XS(2)**2+XS(3)**2
          AM(3,1)=AM(3,1)+XS(3)*XD(1)-XD(3)*XS(1)
          AM(3,2)=AM(3,2)+XD(1)*XD(2)-XS(1)*XS(2)
          AM(3,3)=AM(3,3)+XS(1)**2+XD(2)**2+XS(3)**2
          AM(4,1)=AM(4,1)+XS(1)*XD(2)-XD(1)*XS(2)
          AM(4,2)=AM(4,2)+XD(3)*XD(1)-XS(3)*XS(1)
          AM(4,3)=AM(4,3)+XD(2)*XD(3)-XS(2)*XS(3)
          AM(4,4)=AM(4,4)+XS(1)**2+XS(2)**2+XD(3)**2
        ENDDO
C
        CALL DSYEVR('V','A','L',4,AM,4,0D0,0D0,0,0,0D0,I,EV,RM,4,IV,WV,
     &  132,JV,40,IN)
C        WRITE(6,*) IN,JV(1),NINT(WV(1))
C        STOP
        IF (IN.NE.0) THEN
          WRITE(6,'(/A,I5)') 'Error code:',IN
          CALL CCPERR(1,'ERROR in DSYEVR.')
        ENDIF
        WRITE(6,'(/A,1P/4D12.3/0P,4(/4F12.6))')
     &  'Eigenvalues & vectors:',EV,((RM(I,J),J=1,4),I=1,4)
C
C        WRITE(6,'(/A,F7.3,1X,I7)') 'RMSdev = ',SQRT(EV(1)/(NAF-3)),NAF
C
        S=0.
        DO I=1,4
          S=S+RM(I,1)**2
        ENDDO
        S=SQRT(S)
        WRITE(6,'(/A,F10.6)') 'Norm of quaternion (should be 1) =',S
C
        DO I=1,4
          RM(I,1)=RM(I,1)/S
        ENDDO
C
        S=0.
        DO I=1,3
          V(I)=RM(1+I,1)
          S=S+V(I)**2
        ENDDO
C
C#### Get the Eulerian rotation angles.
        CA31=2.*S
        CA3=1.-CA31
        S=SIGN(SQRT(S),RM(1,1))
        SA3=2.*RM(1,1)*S
        IF (S.NE.0.) THEN
          DO I=1,3
            V(I)=V(I)/S
          ENDDO
        ENDIF
C
        AV(1)=RD*ACOS(V(3))
        IF (V(1).EQ.0. .AND. V(2).EQ.0.) THEN
          AV(2)=0.
        ELSE
          AV(2)=RD*ATAN2(V(2),V(1))
        ENDIF
        AV(3)=RD*ACOS(CA3)
        WRITE(6,'(/A,2X,3F9.2)') 'Polar rotation:',AV
C
C#### Rotation matrix.
        DO I=1,3
          J=MOD(I,3)+1
          K=MOD(J,3)+1
          RM(I,I)=CA31*V(I)**2+CA3
          RM(J,K)=CA31*V(J)*V(K)+SA3*V(I)
          RM(K,J)=CA31*V(J)*V(K)-SA3*V(I)
        ENDDO
C
        DO I=1,3
          TV(I)=XC(I,1)
          DO J=1,3
            TV(I)=TV(I)-RM(I,J)*XC(J,2)
          ENDDO
        ENDDO
C
C#### Fitted co-ords.
```

```
        DO IA2=1,NAP(2)
          DO I=1,3
            S=TV(I)
            DO J=1,3
              S=S+RM(I,J)*XAP(J,IA2,2)
            ENDDO
            XAP(I,IA2,3)=S
          ENDDO
        ENDDO
C
        DM=0.
        DO IAF=1,NAF
          IA1=KAF(IAF,1)
          IA2=KAF(IAF,2)
          D=0.
          DO I=1,3
            D=D+(XAP(I,IA2,3)-XAP(I,IA1,1))**2
          ENDDO
          DM=MAX(DM,D)
C          IF (D.LE.DM) THEN
C            WRITE(6,'(A,F8.3)') ANP(IA1,1),SQRT(D)
C          ELSE
C            DM=D
C            WRITE(6,'(A,F8.3,3X,A)') ANP(IA1,1),SQRT(D),'MAX'
C          ENDIF
        ENDDO
C
        EV(1)=SQRT(EV(1)/(NAF-3))
        WRITE(6,'(2(/A,F7.3))') 'RMSDev = ',EV(1),'MaxDev = ',SQRT(DM)
C
C#### Atom pair distance deviations.
        IF (DMAXS.GT.0.) THEN
          JW=3-IW
          DO IA2=1,NAP(2)
            KAP(IA2,IW)=0
            DM=DMAXS
            DO IA1=1,NAP(1)
              IF (ANP(IA1,1)(:4).EQ.ANP(IA2,2)(:4)) THEN
                D=0.
                DO I=1,3
                  D=D+(XAP(I,IA2,3)-XAP(I,IA1,1))**2
                ENDDO
                IF (D.LT.DM) THEN
                  KAP(IA2,IW)=IA1
                  DM=D
                ENDIF
              ENDIF
            ENDDO
            DAP(IA2)=SQRT(DM)
          ENDDO
C
C8         LP=.TRUE.
8         NCM=0
          DO IA2=1,NAP(2)
            NCP(IA2)=0
            IA1=KAP(IA2,IW)
            IF (IA1.GT.0) THEN
              IR1=INRP(IA1,1)
              IR2=INRP(IA2,2)
              DO JA2=1,NAP(2)
                JA1=KAP(JA2,IW)
                IF (JA1.GT.0) THEN
                  I1=INRP(JA1,1)-IR1
                  IF (I1.NE.0) I1=SIGN(1,I1)
                  I2=INRP(JA2,2)-IR2
                  IF (I2.NE.0) I2=SIGN(1,I2)
                  IF (I2.NE.I1) THEN
C                    WRITE(6,'(4I5)') IR1,IR2,INRP(JA1,1),INRP(JA2,2)
                    NCP(IA2)=NCP(IA2)+1
C                    IF (IT.GT.1) THEN
C                      IF (LP) THEN
C                        WRITE(6,'(/A/)') 'Conflicts:'
C                        LP=.FALSE.
C                      ENDIF
C                      WRITE(6,'(2(4X,A),4X,2(4X,A),I7)') ANP(IA1,1),
C    &                 ANP(IA2,2)(5:),ANP(JA1,1),ANP(JA2,2)(5:),NCP(IA2)
C                    ENDIF
                  ENDIF
                ENDIF
              ENDDO
              NCM=MAX(NCM,NCP(IA2))
           ENDIF
         ENDDO
C
         IF (NCM.GT.0) THEN
C           IF (IT.GT.1) WRITE(6,'()')
           DO IA2=1,NAP(2)
C             IF (NCP(IA2).EQ.NCM) THEN
C               IA1=KAP(IA2,IW)
C               IF (IT.GT.1) WRITE(6,'(A,2(4X,A),I7)') 'Reject',
C    &          ANP(IA1,1),ANP(IA2,2)(5:),NCM
C               KAP(IA2,IW)=0
C             ENDIF
             IF (NCP(IA2).EQ.NCM) KAP(IA2,IW)=0
           ENDDO
           GOTO 8
         ENDIF
C
C         WRITE(6,'(/A/)') 'Alignment:'
C         DO IA2=1,NAP(2)
C           IA1=KAP(IA2,IW)
C           IF (IA1.GT.0) WRITE(6,'(2(I5,2X,A,2X),F8.3)') IA1,
C    &      ANP(IA1,1),IA2,ANP(IA2,2),DAP(IA2)
C         ENDDO
C
         IRP=1
         JRP(IRP)=0
         DO I=1,3
           LA(I)=.FALSE.
         ENDDO
C
         DO IA2=1,NAP(2)
           IA1=KAP(IA2,IW)
           IF (IA1.GT.0) THEN
             DO I=1,3
               IF (ANP(IA2,2)(:2).EQ.AA(I)) THEN
                 LA(I)=.TRUE.
                 JRP(IRP)=INRP(IA1,1)
                 GOTO 15
               ENDIF
             ENDDO
           ENDIF
```

```
15          IF (INRP(IA2+1,2).GT.IRP) THEN
              LRP(1,IRP)=LA(1).AND.LA(2)
              LRP(2,IRP)=LA(2).AND.LA(3)
              IRP=IRP+1
              JRP(IRP)=0
              DO I=1,3
                LA(I)=.FALSE.
              ENDDO
            ENDIF
          ENDDO
C
C          IF (IT.GT.1) WRITE(6,'()')
          IF (.NOT.(LRP(2,1) .AND. LRP(1,2)) .AND. JRP(2).EQ.JRP(1)+1)
     &    THEN
            LRP(2,1)=.FALSE.
            JRP(1)=0
            DO IA2=1,IRAP(1,2)
C              IA1=KAP(IA2,IW)
C              IF (IA1.GT.0) THEN
C                KAP(IA2,IW)=0
C                IF (IT.GT.1) WRITE(6,'(A,2(4X,A))') 'Reject',
C     &          ANP(IA1,1),ANP(IA2,2)(5:)
C              ENDIF
              KAP(IA2,IW)=0
            ENDDO
          ENDIF
C
          DO IRP=2,NRP(2)-1
            IF (.NOT.(LRP(2,IRP-1) .AND. LRP(1,IRP) .OR. LRP(2,IRP)
     &      .AND. LRP(1,IRP+1))) THEN
              LRP(1,IRP)=.FALSE.
              LRP(2,IRP)=.FALSE.
              JRP(IRP)=0
              DO IA2=IRAP(IRP-1,2)+1,IRAP(IRP,2)
C                IA1=KAP(IA2,IW)
C                IF (IA1.GT.0) THEN
C                  KAP(IA2,IW)=0
C                  IF (IT.GT.1) WRITE(6,'(A,2(4X,A))') 'Reject',
C     &            ANP(IA1,1),ANP(IA2,2)(5:)
C                ENDIF
                KAP(IA2,IW)=0
              ENDDO
            ENDIF
          ENDDO
C
          IF (.NOT.(LRP(2,NRP(2)-1) .AND. LRP(1,NRP(2)))) THEN
            LRP(1,NRP(2))=.FALSE.
            JRP(NRP(2))=0
            DO IA2=IRAP(NRP(2)-1,2)+1,IRAP(NRP(2),2)
C              IA1=KAP(IA2,IW)
C              IF (IA1.GT.0) THEN
C                KAP(IA2,IW)=0
C                IF (IT.GT.1) WRITE(6,'(A,2(4X,A))') 'Reject',
C     &          ANP(IA1,1),ANP(IA2,2)(5:)
C              ENDIF
              KAP(IA2,IW)=0
            ENDDO
          ENDIF
C
C          WRITE(6,'(/A/)') 'Alignment:'
C          DO IA2=1,NAP(2)
```

```
C            IA1=KAP(IA2,IW)
C            IF (IA1.GT.0) WRITE(6,'(2(I5,2X,A,2X),F8.3)') IA1,
C     &      ANP(IA1,1),IA2,ANP(IA2,2),DAP(IA2)
C          ENDDO
C
C          IF (IT.GT.1) THEN
C            LP=.TRUE.
C            DO IA2=1,NAP(2)
C              IA1=KAP(IA2,IW)
C              JA1=KAP(IA2,JW)
C              IF (IA1.NE.JA1) THEN
C                IF (LP) THEN
C                  WRITE(6,'(/A/)') 'Changes in alignment:'
C                  LP=.FALSE.
C                ENDIF
C                A(:20)=' '
C                IF (JA1.GT.0) A(:10)=ANP(JA1,1)(5:)
C                IF (IA1.GT.0) A(11:20)=ANP(IA1,1)(5:)
C                WRITE(6,'(A,3(4X,A))') ANP(IA2,2)(:4),A(:10),A(11:20),
C     &          ANP(IA2,2)(5:)
C              ENDIF
C            ENDDO
C          ENDIF
C
          IF (IT.LT.MIT) THEN
C            WRITE(6,'()')
            DO IA2=1,NAP(2)
              IF (KAP(IA2,IW).NE.KAP(IA2,JW)) GOTO 12
            ENDDO
            GOTO 13
C
12          NAF=0
            DO IA2=1,NAP(2)
              IF (KAP(IA2,IW).GT.0) THEN
                NAF=NAF+1
                KAF(NAF,1)=KAP(IA2,IW)
                KAF(NAF,2)=IA2
              ENDIF
            ENDDO
            IF (NAF.LE.3) THEN
              CALL ERRMSG('ERROR: No fit!')
              NS=NS-1
              GOTO 1
            ENDIF
            IW=JW
            GOTO 9
          ENDIF
          CALL ERRMSG('ERROR: Not converged.')
          NS=NS-1
          GOTO 1
        ENDIF
C
C#### Write out final alignment.
13      IF (DMAXS.GT.0.) THEN
          NAT=0
          NAI=0
          NAF=0
          D=0.
          DO JA2=1,NAP(2)
            IF (ANP(JA2,2)(:3).EQ.'CA') THEN
              JA1=KAP(JA2,IW)
              IF (JA1.GT.0) GOTO 11
            ENDIF
          ENDDO
          JA1=NAP(1)+1
11        DO I=1,JA1-1
            IF (ANP(I,1)(:3).EQ.'CA') THEN
              IF (NAT.EQ.0) WRITE(6,'(//A/)')
     &        'Final structure-based sequence alignment:'
              NAT=NAT+1
              WRITE(6,'(I4,4X,A)') NAT,ANP(I,1)(5:)
            ENDIF
          ENDDO
          DO IA2=1,NAP(2)
            IF (ANP(IA2,2)(:3).EQ.'CA') THEN
              IA1=KAP(IA2,IW)
              IF (IA1.EQ.0) THEN
                NAT=NAT+1
                WRITE(6,'(I4,26X,A)') NAT,ANP(IA2,2)(5:)
              ELSE
                IF (NAT.EQ.0) WRITE(6,'(//A/)')
     &          'Final structure-based sequence alignment:'
                NAT=NAT+1
                IF (ANP(IA1,1)(5:7).EQ.ANP(IA2,2)(5:7)) THEN
                  NAI=NAI+1
                  WRITE(6,'(I4,I8,4X,2(A,4X),F6.3)') NAT,NAI,
     &            ANP(IA1,1)(5:),ANP(IA2,2)(5:),DAP(IA2)
                ELSE
                  WRITE(6,'(I4,12X,2(A,4X),F6.3)') NAT,ANP(IA1,1)(5:),
     &            ANP(IA2,2)(5:),DAP(IA2)
                ENDIF
                DO JA2=IA2+1,NAP(2)
                  IF (ANP(JA2,2)(:3).EQ.'CA') THEN
                    JA1=KAP(JA2,IW)
                    IF (JA1.GT.0) GOTO 14
                  ENDIF
                ENDDO
                JA1=NAP(1)+1
14              DO I=IA1+1,JA1-1
                  IF (ANP(I,1)(:3).EQ.'CA') THEN
                    NAT=NAT+1
                    WRITE(6,'(I4,12X,A)') NAT,ANP(I,1)(5:)
                  ENDIF
                ENDDO
C
                NAF=NAF+1
                DO I=1,3
                  D=D+(XAP(I,IA2,3)-XAP(I,IA1,1))**2
                ENDDO
              ENDIF
            ENDIF
          ENDDO
          IF (2*NAF.LE.MIN(NRP(1),NRP(2))) THEN
            CALL ERRMSG('ERROR: No fit!')
            NS=NS-1
            GOTO 1
          ENDIF
          WRITE(6,'(/2(A,I5))') 'No of identical residues = ',NAI,
     &    ' out of ',NAT
          WRITE(6,'(/A,F6.3,A,I4,A,F6.3)') 'Structural identity = ',
     &    REAL(2*NAI)/(NRP(1)+NRP(2)),' ,  RMSDev for ',NAF,
     &    ' CA atoms = ',SQRT(D/(NAF-3))
```

```
        ENDIF
C
C#### Write output PDB file.
        IF (LDIGA(1,B).GT.0) THEN
          OPEN(2,FILE=B,STATUS='UNKNOWN')
          DO I=1,3
            XL(I)=1E38
            XM(I)=-1E38
          ENDDO
C
4         READ(1,'(A)',END=5) A
          IF ((A(:6).EQ.'ATOM' .OR. A(:6).EQ.'HETATM') .AND.
     &    A(18:20).NE.'HOH' .AND. A(18:20).NE.'WAT' .AND.
     &    (CSF(2).EQ.' ' .OR. INDEX(CSF(2),A(22:22)).GT.0)) THEN
            READ(A(31:54),'(3F8.3)') X1
            DO I=1,3
              X2(I)=TV(I)
              DO J=1,3
                X2(I)=X2(I)+RM(I,J)*X1(J)
              ENDDO
            ENDDO
            WRITE(A(31:54),'(3F8.3)') X2
            DO I=1,3
              XL(I)=MIN(XL(I),X2(I))
              XM(I)=MAX(XM(I),X2(I))
            ENDDO
C
            IF (IBC.LT.0) THEN
              DM=1E38
              DO IA1=1,NAP(1)
                IF (ANP(IA1,1)(:2).EQ.A(14:15)) THEN
                  D=0.
                  DO I=1,3
                    D=D+(X2(I)-XAP(I,IA1,1))**2
                  ENDDO
                  DM=MIN(DM,D)
                ENDIF
              ENDDO
              WRITE(A(61:66),'(F6.2)') MIN(SQRT(DM),999.99)
            ELSEIF (IBC.GT.0) THEN
              WRITE(A(61:66),'(F6.2)') BF
            ENDIF
            WRITE(2,'(A)') A(:LENSTR(A))
          ENDIF
          GOTO 4
5         CLOSE(2)
          REWIND 1
          WRITE(6,'(3(/A,3F9.3))') 'Extent (min)    :',XL,
     &    'Extent (max)    :',XM,'Extent (total) :',(XM(I)-XL(I),I=1,3)
        ENDIF
C
C#### Write transformation matrix, Eulerian angles, orthogonal
C#### translation.
        AV(2)=RD*ACOS(RM(3,3))
        IF (RM(1,3).EQ.0. .AND. RM(2,3).EQ.0. .OR. RM(3,1).EQ.0. .AND.
     &  RM(3,2).EQ.0.) THEN
          AV(1)=0.
          AV(3)=RD*ATAN2(RM(2,1),RM(1,1))
        ELSE
          AV(1)=RD*ATAN2(RM(2,3),RM(1,3))
          IF (AV(1).LT.0.) AV(1)=AV(1)+360.
          AV(3)=RD*ATAN2(RM(3,2),-RM(3,1))
        ENDIF
        IF (AV(3).LT.0.) AV(3)=AV(3)+360.
        WRITE(6,'(/A,3(/3F9.4,F10.3))') 'Transformation matrix:',
     &  ((RM(I,J),J=1,3),TV(I),I=1,3)
        WRITE(6,'(/A,2X,3F9.2/A,2X,3F9.3/)') 'Eulerian rotation:     ',
     &  AV,'Orthogonal translation:',TV
C
C#### Transform another file with the same matrix.
      ELSEIF (L.GE.4 .AND. B.EQ.'TRANSFORM_MOL'(:L)) THEN
        IF (NS.LT.2) THEN
          CALL ERRMSG('ERROR: Need 2 or more files.')
          GOTO 1
        ENDIF
        IF (LDIGA(2,BA).EQ.0) THEN
          CALL ERRMSG('ERROR: File name missing.')
          GOTO 1
        ENDIF
        OPEN(3,FILE=BA(1),STATUS='OLD')
        OPEN(2,FILE=BA(2),STATUS='UNKNOWN')
        READ(3,'(///2I3)',ERR=17) NA,NB
        REWIND 3
        DO IA1=1,4
          READ(3,'(A)') A
          WRITE(2,'(A)') A(:LENSTR(A))
        ENDDO
        DO IA1=1,NA
          READ(3,'(3F10.4,A)') X1,A
          DO I=1,3
            X2(I)=TV(I)
            DO J=1,3
              X2(I)=X2(I)+RM(I,J)*X1(J)
            ENDDO
          ENDDO
          WRITE(2,'(3F10.4,A)') X2,A(:LENSTR(A))
        ENDDO
        DO IA1=1,NB
          READ(3,'(A)') A
          WRITE(2,'(A)') A(:LENSTR(A))
        ENDDO
16      READ(3,'(A)',END=19) A
        WRITE(2,'(A)') A(:LENSTR(A))
        IF (A(:6).EQ.'M  END') GOTO 19
        GOTO 16
17      REWIND 3
18      READ(3,'(A)',END=19) A
        IF (A(:6).EQ.'ATOM' .OR. A(:6).EQ.'HETATM') THEN
          READ(A(31:54),'(3F8.3)') X1
          DO I=1,3
            X2(I)=TV(I)
            DO J=1,3
              X2(I)=X2(I)+RM(I,J)*X1(J)
            ENDDO
          ENDDO
          WRITE(A(31:54),'(3F8.3)') X2
        ENDIF
        WRITE(2,'(A)') A(:LENSTR(A))
        GOTO 18
19      CLOSE(2)
        CLOSE(3)
        GOTO 1
      ELSE
        CALL ERRMSG('ERROR: Invalid input: '//A(:LENSTR(A)))
      ENDIF
      GOTO 1
6     END
C
C
      SUBROUTINE ERRMSG(A)
      IMPLICIT NONE
      CHARACTER A*(*)
      IF (.NOT.ISATTY(6)) CALL CCPERR(1,A)
      WRITE(6,'(/A)') A
      END
C
C
      SUBROUTINE SEQALG(SMAX,NRP,RNP,RTP,NRF,KRF)
C#### Needleman-Wunsch-Sellers sequence alignment using BLOSUM62
C#### substitution matrix & gap penalties.
C
      IMPLICIT NONE
      CHARACTER RT1*3,RT2*3,RN1*8,RN2*8
      LOGICAL LG
      INTEGER I,I1,I2,IGE,IGO,IRP,IS,ISA,ITE,ITO,J1,J2,JSA,MRP,MSA,MSUM,
     &N1,N2,NHW,NRP1,NRP2,NSA
      REAL S,SC,SMAX
      PARAMETER (MRP=4000,MSA=2*MRP,MSUM=20,NHW=9)
      INTEGER KRF(MRP,2),KSUM(0:MSUM,0:MSUM),KSA(2,MSA),KTP(MRP,2),
     &NRF(2),NRP(2),LA(2)
      CHARACTER RNP(MRP,2)*8,RTA(MSUM)*3,RTP(MRP,2)*3,TA(2)
      REAL P(-NHW:NHW),SA(2,MSA)
      COMMON/SEQCOM/ KSUM,IGO,IGE,ITO,ITE,NRP1,NRP2,KTP,NSA,KSA
      EXTERNAL NWSALG
      DATA TA/2*'I'/
C
C#### List of a.a. residues and BLOSUM62 substitution matrix.
      DATA RTA/'ALA','ARG','ASN','ASP','CYS','GLN','GLU','GLY','HIS',
     &'ILE','LEU','LYS','MET','PHE','PRO','SER','THR','TRP','TYR','VAL'/
      DATA KSUM/
     & 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
     & 0, 4,-1,-2,-2, 0,-1,-1, 0,-2,-1,-1,-1,-1,-2,-1, 1, 0,-3,-2, 0,
     & 0,-1, 5, 0,-2,-3, 1, 0,-2, 0,-3,-2, 2,-1,-3,-2,-1,-1,-3,-2,-3,
     & 0,-2, 0, 6, 1,-3, 0, 0, 0, 1,-3,-3, 0,-2,-3,-2, 1, 0,-4,-2,-3,
     & 0,-2,-2, 1, 6,-3, 0, 2,-1,-1,-3,-4,-1,-3,-3,-1, 0,-1,-4,-3,-3,
     & 0, 0,-3,-3,-3, 9,-3,-4,-3,-3,-1,-1,-3,-1,-2,-3,-1,-1,-2,-2,-1,
     & 0,-1, 1, 0, 0,-3, 5, 2,-2, 0,-3,-2, 1, 0,-3,-1, 0,-1,-2,-1,-2,
     & 0,-1, 0, 0, 2,-4, 2, 5,-2, 0,-3,-3, 1,-2,-3,-1, 0,-1,-3,-2,-2,
     & 0, 0,-2, 0,-1,-3,-2,-2, 6,-2,-4,-4,-2,-3,-3,-2, 0,-2,-2,-3,-3,
     & 0,-2, 0, 1,-1,-3, 0, 0,-2, 8,-3,-3,-1,-2,-1,-2,-1,-2,-2, 2,-3,
     & 0,-1,-3,-3,-3,-1,-3,-3,-4,-3, 4, 2,-3, 1, 0,-3,-2,-1,-3,-1, 3,
     & 0,-1,-2,-3,-4,-1,-2,-3,-4,-3, 2, 4,-2, 2, 0,-3,-2,-1,-2,-1, 1,
     & 0,-1, 2, 0,-1,-3, 1, 1,-2,-1,-3,-2, 5,-1,-3,-1, 0,-1,-3,-2,-2,
     & 0,-1,-1,-2,-3,-1, 0,-2,-3,-2, 1, 2,-1, 5, 0,-2,-1,-1,-1,-1, 1,
     & 0,-2,-3,-3,-3,-2,-3,-3,-3,-1, 0, 0,-3, 0, 6,-4,-2,-2, 1, 3,-1,
     & 0,-1,-2,-2,-1,-3,-1,-1,-2,-2,-3,-3,-1,-2,-4, 7,-1,-1,-4,-3,-2,
     & 0, 1,-1, 1, 0,-1, 0, 0, 0,-1,-2,-2, 0,-1,-2,-1, 4, 1,-3,-2,-2,
     & 0, 0,-1, 0,-1,-1,-1,-1,-2,-2,-1,-1,-1,-1,-2,-1, 1, 5,-2,-2, 0,
     & 0,-3,-3,-4,-4,-2,-2,-3,-2,-2,-3,-2,-3,-1, 1,-4,-3,-2,11, 2,-3,
     & 0,-2,-2,-2,-3,-2,-1,-2,-3, 2,-1,-1,-2,-1, 3,-3,-2,-2, 2, 7,-1,
     & 0, 0,-3,-3,-3,-1,-2,-2,-3,-3, 3, 1,-2, 1,-1,-2,-2, 0,-3,-1, 4/
C
C#### Gap penalties and score cutoff.
```

```
      DATA IGO,IGE,ITO,ITE/12,1,1,1/, SC/2.5/
C
C#### Test for difference in sequences.
      WRITE(6,'(/A,2I6)') 'No of residues =',NRP
      IF (NRP(1).EQ.NRP(2)) THEN
        DO IRP=1,NRP(1)
          IF (RTP(IRP,2).NE.RTP(IRP,1)) THEN
            WRITE(6,'(/A,2(4X,A,2X,A))') 'Sequences differ at',
     &      (RTP(IRP,IS),RNP(IRP,IS),IS=1,2)
            GOTO 1
          ENDIF
        ENDDO
C
C#### Sequences are identical, skip sequence alignment.
        DO IS=1,2
          NRF(IS)=NRP(1)
          DO IRP=1,NRP(1)
            KRF(IRP,IS)=IRP
          ENDDO
        ENDDO
        WRITE(6,'(/A/)')
     &  'Sequences are identical, skipping sequence alignment.'
        IF (SMAX.LT.1.) CALL CCPERR(0,'TERMINATED - IDENTITY')
        RETURN
      ENDIF
C
C#### Set up residue type pointers.
1     DO IS=1,2
        DO IRP=1,NRP(IS)
          DO I=1,MSUM
            IF (RTP(IRP,IS).EQ.RTA(I)) GOTO 2
          ENDDO
          I=0
2         KTP(IRP,IS)=I
        ENDDO
      ENDDO
C
C#### Allocate memory for scoring matrix.
      NRP1=NRP(1)
      NRP2=NRP(2)
      LA(1)=(1+NRP1)*(1+NRP2)
      LA(2)=LA(1)
      CALL CCPALC(NWSALG,2,TA,LA)
C
C#### Compute smoothed out similarity scores.
      I1=0
      I2=0
      N1=0
      N2=0
      DO ISA=1,NSA
        J1=KSA(1,ISA)
        IF (J1.GT.0) I1=I1+1
        J2=KSA(2,ISA)
        IF (J2.GT.0) I2=I2+1
C
        IF (J1.EQ.0) THEN
C            IF (J2.GT.0) WRITE(6,'(13X,A,1X,A)') RTA(J2),RNP(I2,2)
          N1=N1+1
        ELSEIF (J2.EQ.0) THEN
C            IF (J1.GT.0) WRITE(6,'(A,1X,A)') RTA(J1),RNP(I1,1)
          N2=N2+1
        ELSE
C          WRITE(6,'(A,1X,A,3X,A,1X,A,I5)') RTA(J1),RNP(I1,1),RTA(J2),
C     &    RNP(I2,2),KSUM(J1,J2)
          SA(1,ISA)=KSUM(J1,J2)
        ENDIF
C
        IF (J1.GT.0 .AND. N1.GT.0) THEN
          IF (ISA.EQ.N1+1) THEN
            S=-REAL(ITO+(N1-1)*ITE)/N1
          ELSE
            S=-REAL(IGO+(N1-1)*IGE)/N1
          ENDIF
          DO JSA=ISA-N1,ISA-1
            SA(1,JSA)=S
          ENDDO
          N1=0
        ENDIF
C
        IF (J2.GT.0 .AND. N2.GT.0) THEN
          IF (ISA.EQ.N2+1) THEN
            S=-REAL(ITO+(N2-1)*ITE)/N2
          ELSE
            S=-REAL(IGO+(N2-1)*IGE)/N2
          ENDIF
          DO JSA=ISA-N2,ISA-1
            SA(1,JSA)=S
          ENDDO
          N2=0
        ENDIF
      ENDDO
C
      IF (N1.GT.0) THEN
        S=-REAL(ITO+(N1-1)*ITE)/N1
        DO JSA=NSA+1-N1,NSA
          SA(1,JSA)=S
        ENDDO
      ENDIF
C
      IF (N2.GT.0) THEN
        S=-REAL(ITO+(N2-1)*ITE)/N2
        DO JSA=NSA+1-N2,NSA
          SA(1,JSA)=S
        ENDDO
      ENDIF
C
      S=.25**NHW
      P(NHW)=S
      DO I=NHW-1,0,-1
        P(I)=P(I+1)*(NHW+I+1)/(NHW-I)
      ENDDO
      DO I=-NHW,-1
        P(I)=P(-I)
      ENDDO
C
C#### Write out the sequence alignment & similarity scores.
      WRITE(6,'(/A)') 'Sequence alignment with similarity scores:'
      LG=.TRUE.
      I1=0
      I2=0
      NRF(1)=0
      DO ISA=1,NSA
```

```
        J1=KSA(1,ISA)
        IF (J1.EQ.0) THEN
          RN1='  '
          RT1='  '
        ELSE
          I1=I1+1
          RN1=RNP(I1,1)
          RT1=RTA(J1)
        ENDIF
C
        J2=KSA(2,ISA)
        IF (J2.EQ.0) THEN
          RN2='  '
          RT2='  '
        ELSE
          I2=I2+1
          RN2=RNP(I2,2)
          RT2=RTA(J2)
        ENDIF
C
        SA(2,ISA)=0.
        DO JSA=MAX(ISA-NHW,1),MIN(ISA+NHW,NSA)
          SA(2,ISA)=SA(2,ISA)+P(JSA-ISA)*SA(1,JSA)
        ENDDO
C
        IF (J1.EQ.0 .OR. J2.EQ.0 .OR. SA(2,ISA).LE.SC) THEN
          LG=.TRUE.
        ELSE
          IF (LG) THEN
            LG=.FALSE.
            WRITE(6,'()')
          ENDIF
C          WRITE(6,'(A,2X,A,2X,A,2X,A,2F5.1)') RT1,RN1,RT2,RN2,
C     &    SA(1,ISA),SA(2,ISA)
          WRITE(6,'(A,2X,A,2X,A,2X,A,F5.1)') RT1,RN1,RT2,RN2,SA(2,ISA)
          NRF(1)=NRF(1)+1
          KRF(NRF(1),1)=I1
          KRF(NRF(1),2)=I2
        ENDIF
      ENDDO
      NRF(2)=NRF(1)
      S=REAL(NRF(1))/NSA
      WRITE(6,'(/A,2I6,F7.3/)') 'Sequence identity =',NSA,NRF(1),S
      IF (S.GT.SMAX) CALL CCPERR(0,'TERMINATED - IDENTITY')
      END
C
C
      SUBROUTINE NWSALG(LA1,KGPM,LA2,KSCM)
C#### Needleman-Wunsch-Sellers sequence alignment.
C
      IMPLICIT NONE
      INTEGER I,I1,I2,IGE,IGO,IS,ISA,ISC,ISC1,ISC2,ITE,ITO,J,J1,J2,JSA,
     &K1,K2,LA1,LA2,MRP,MSA,MSUM,NRP1,NRP2,NSA
      PARAMETER (MRP=4000,MSA=2*MRP,MSUM=20)
      INTEGER KGPM(0:NRP1,0:NRP2),KSCM(0:NRP1,0:NRP2),KSA(2,MSA),
     &KSUM(0:MSUM,0:MSUM),KTP(MRP,2),NRP(2)
      COMMON/SEQCOM/ KSUM,IGO,IGE,ITO,ITE,NRP1,NRP2,KTP,NSA,KSA
      EQUIVALENCE(NRP1,NRP)
C
      DO IS=1,2
        DO I=1,NRP(IS)
```

```
          IF (KTP(I,IS).LT.0 .OR. KTP(I,IS).GT.MSUM) THEN
            WRITE(6,'(3I5)') I,KTP(I,IS),MSUM
            CALL CCPERR(1,'ERROR: Code out of range.')
          ENDIF
        ENDDO
      ENDDO
C
C#### Initial matrix elements with gap penalties.
      KGPM(0,0)=0
      KSCM(0,0)=0
      KGPM(1,0)=1
      KSCM(1,0)=ITO
      KGPM(0,1)=2
      KSCM(0,1)=ITO
      DO I1=2,NRP1
        KGPM(I1,0)=1
        KSCM(I1,0)=KSCM(I1-1,0)-ITE
      ENDDO
      DO I2=2,NRP2
        KGPM(0,I2)=2
        KSCM(0,I2)=KSCM(0,I2-1)-ITE
      ENDDO
C
C#### Accumulate matrix elements.
      DO I2=1,NRP2
        J2=I2-1
        K2=KTP(I2,2)
C        WRITE(6,'()')
        DO I1=1,NRP1
          J1=I1-1
          K1=KTP(I1,1)
          KGPM(I1,I2)=0
          ISC=KSCM(J1,J2)+KSUM(K1,K2)
          IF (I2.LT.NRP2) THEN
            IF (KGPM(J1,I2).NE.1) THEN
              ISC1=KSCM(J1,I2)-IGO
            ELSE
              ISC1=KSCM(J1,I2)-IGE
            ENDIF
          ELSE
            IF (KGPM(J1,I2).NE.1) THEN
              ISC1=KSCM(J1,I2)-ITO
            ELSE
              ISC1=KSCM(J1,I2)-ITE
            ENDIF
          ENDIF
          IF (ISC1.GE.ISC) THEN
            KGPM(I1,I2)=1
            ISC=ISC1
          ENDIF
          IF (I1.LT.NRP1) THEN
            IF (KGPM(I1,J2).NE.2) THEN
              ISC2=KSCM(I1,J2)-IGO
            ELSE
              ISC2=KSCM(I1,J2)-IGE
            ENDIF
          ELSE
            IF (KGPM(I1,J2).NE.2) THEN
              ISC2=KSCM(I1,J2)-ITO
            ELSE
              ISC2=KSCM(I1,J2)-ITE
            ENDIF
          ENDIF
          IF (ISC2.GE.ISC) THEN
            KGPM(I1,I2)=2
            ISC=ISC2
          ENDIF
          KSCM(I1,I2)=ISC
C          WRITE(6,'(2I3,2X,3I3,2X,2I3,2X,2I3,2X,2I3,2X,I3)') I1,I2,K1,
C    &     K2,KSUM(K1,K2),KSCM(J1,J2),KSCM(J1,J2)+KSUM(K1,K2),
C    &     KSCM(J1,I2),ISC1,KSCM(I1,J2),ISC2,KGPM(I1,I2)
        ENDDO
      ENDDO
C
C      DO I2=0,NRP2
C        WRITE(6,'()')
C        WRITE(6,'(20I5)') (KGPM(I1,I2),I1=0,NRP1)
C        WRITE(6,'(20I5)') (KSCM(I1,I2),I1=0,NRP1)
C      ENDDO
      WRITE(6,'()')
C
C#### Find the optimal path through the matrix.
      I1=NRP1
      I2=NRP2
      NSA=0
1     IF (NSA.EQ.MSA) CALL CCPERR(1,'Increase MRP.')
      NSA=NSA+1
C      WRITE(6,'(2I5,I4,I6)') I1,I2,KGPM(I1,I2),KSCM(I1,I2)
      IF (KGPM(I1,I2).EQ.0) THEN
        KSA(1,NSA)=KTP(I1,1)
        KSA(2,NSA)=KTP(I2,2)
        I1=I1-1
        I2=I2-1
      ELSEIF (KGPM(I1,I2).EQ.1) THEN
        KSA(1,NSA)=KTP(I1,1)
        KSA(2,NSA)=0
        I1=I1-1
      ELSE
        KSA(1,NSA)=0
        KSA(2,NSA)=KTP(I2,2)
        I2=I2-1
      ENDIF
C      WRITE(6,'(20X,I6,2I4)') NSA,(KSA(I,NSA),I=1,2)
      IF (I1.GT.0 .OR. I2.GT.0) GOTO 1
C      WRITE(6,'()')
C
C#### Reverse the order of the residue pointers.
      JSA=NSA
      DO ISA=1,NSA/2
        DO I=1,2
          J=KSA(I,ISA)
          KSA(I,ISA)=KSA(I,JSA)
          KSA(I,JSA)=J
        ENDDO
        JSA=JSA-1
      ENDDO
      END
C
C
      INCLUDE'ldigr.f'
```

ANNEX 5

```
      INTEGER FUNCTION ldigr(LUN,STR)
C
C#### List-directed input library.
C
C#### Ian J. Tickle, Astex Technology.
C#### Copyright © 1980-2003 Ian J. Tickle.
C
C#### This library is free software; you can redistribute it and/or
C#### modify it under the terms of the GNU Library General Public
C#### License as published by the Free Software Foundation; either
C#### version 2 of the License, or (at your option) any later version.
C
C#### This library is distributed in the hope that it will be useful,
C#### but WITHOUT ANY WARRANTY; without even the implied warranty of
C#### MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
C#### Library General Public License for more details:
C#### http://www.lesstif.org/COPYING.LIB.html
C
C#### List-directed input, get record.
C
C#### LUN = Logical unit for read.
C#### STR = Record string read; length should be max record length + 1.
C#### Function value returned = -1 for end-of-file.
C####                         =  0 for null record.
C####                         =  Record length in bytes.
C
      IMPLICIT NONE
      CHARACTER STR*(*),AA(*)*(*),REC*2001,C,NUM*10
      LOGICAL F,FP,FE
      INTEGER LUN,LR,N,N0,NA,IA,LA,NT,IT,NI,I,J,IH,IS,NF,IE,JE
      INTEGER II(*),LP(*),MP(*)
      INTEGER LENSTR,LDISR,LDIGA,LDIGC,LDIGI,LDIGF,LDIGP,LDIGT
      REAL FF(*)
      DOUBLEPRECISION DF
      CHARACTER SUBSTR
      SAVE LR,N,REC
      DATA NUM/'0123456789'/
      DATA DF/1D0/
C
      I=1
1     READ (LUN,'(A)',END=2) REC(I:)
      LR=LENSTR(REC)
C#### Check for continuation (- or \).
      IF (LR.GT.0) THEN
        IF (REC(LR:LR).EQ.'-' .OR. REC(LR:LR).EQ.SUBSTR('\\',1)) THEN
          I=LR
          GOTO 1
        ENDIF
      ENDIF
      STR=REC
      LR=MIN(LR+1,LEN(REC))
C#### Check for comment (! OR #).
      I=INDEX(REC(:LR),'!')
      J=INDEX(REC(:LR),'#')
      IF (I.EQ.0) THEN
        I=J
      ELSEIF (J.GT.0) THEN
```

```
          I=MIN(I,J)
        ENDIF
        IF (I.GT.0) LR=I
C#### Supply record delimiter.
        REC(LR:LR)=' '
C#### Point to next byte.
        N=1
        LDIGR=LR-1
        RETURN
C#### End of file.
2       LR=0
        LDIGR=-1
        RETURN
C
C
        ENTRY LDISR(STR)
C#### List-directed input, setup record.
C#### STR = Record read; length should be max record length + 1.
C#### Function value returned =  0 for null record.
C####                         =  Record length in bytes.
C
        REC=STR
        LR=MIN(LENSTR(REC)+1,LEN(REC))
C#### Check for comment (! OR #).
        I=INDEX(REC(:LR),'!')
        J=INDEX(REC(:LR),'#')
        IF (I.EQ.0) THEN
          I=J
        ELSEIF (J.GT.0) THEN
          I=MIN(I,J)
        ENDIF
C
        IF (I.GT.0) LR=I
C#### Supply record delimiter.
        REC(LR:LR)=' '
C#### Point to next byte.
        N=1
        LDISR=LR-1
        RETURN
C
C
        ENTRY LDIGA(NA,AA)
C#### List-directed input, get alphanumeric(s).
C#### NA = Number of character elements required.
C#### AA = array to receive NA elements.
C#### Function value returned is the number of elements actually found
C#### in the current record (only 1 record is searched), or NA whichever
C#### is less.
C#### Elements are returned left-aligned.
C#### 1 or more spaces, tabs, comma and apostrophe are delimiters.
C#### Apostrophes in the input string must be doubled and the whole
C#### string enclosed by apostrophes.
C#### A comma following a delimiter skips the element.
C
        LA=LEN(AA(1))
        IA=0
        IF (LA.EQ.0 .OR. NA.LE.0) GOTO 17
C
        F=.FALSE.
11      I=0
C
```

```
C#### Test for end-of-record.
12    IF (N.GT.LR) GOTO 17
C#### Get next char.
      C=REC(N:N)
      N=N+1
      IF (.NOT.F) THEN
        IF (C.EQ.' ' .OR. C.EQ.CHAR(9)) THEN
C#### Space or tab.
          IF (I.EQ.0) GOTO 12
          GOTO 15
C
        ELSEIF (C.EQ.',') THEN
C#### Comma.
          IF (I.EQ.0) THEN
C#### Non-terminating comma skips.
            IA=IA+1
            GOTO 16
          ENDIF
          GOTO 15
        ENDIF
      ENDIF
C
      IF ((F .AND. N.GT.LR) .OR. C.EQ.'''') THEN
C#### Apostrophe.
        IF (.NOT.F) THEN
          F=.TRUE.
          IF (I.EQ.0) GOTO 12
          GOTO 15
        ELSE
          IF (N.LE.LR .AND. REC(N:N).EQ.'''') THEN
            N=N+1
          ELSE
            F=.FALSE.
            IF (I.EQ.0) IA=IA+1
            GOTO 15
          ENDIF
        ENDIF
      ENDIF
C
      IF (I.LT.LA) THEN
        IF (I.EQ.0) IA=IA+1
        I=I+1
C#### Store character.
        AA(IA)(I:I)=C
      ENDIF
      GOTO 12
C
C#### Pad out with spaces.
15    IF (I.LT.LA) AA(IA)(I+1:)=' '
16    IF (IA.LT.NA) GOTO 11
C
17    LDIGA=IA
      RETURN
C
C
      ENTRY LDIGC(NA,AA)
C#### List-directed input, get alphanumeric(s).
C#### NA = Number of character elements required.
C#### AA = array to receive NA elements.
C#### Function value returned is the number of elements actually found
C#### in the current record (only 1 record is searched), or NA whichever
```

```
C#### is less.
C#### Elements are returned left-aligned.
C#### 1 or more spaces, tabs and apostrophe are delimiters.
C#### Apostrophes in the input string must be doubled and the whole
C#### string enclosed by apostrophes.
C#### A comma following a delimiter skips the element.
C
      LA=LEN(AA(1))
      IA=0
      IF (LA.EQ.0 .OR. NA.LE.0) GOTO 6
C
      F=.FALSE.
3     I=0
C
C#### Test for end-of-record.
4     IF (N.GT.LR) GOTO 6
C#### Get next char.
      C=REC(N:N)
      N=N+1
C#### Space or tab.
      IF (.NOT.F .AND. (C.EQ.' ' .OR. C.EQ.CHAR(9))) THEN
        IF (I.EQ.0) GOTO 4
        GOTO 5
      ENDIF
C
      IF ((F .AND. N.GT.LR) .OR. C.EQ.'''') THEN
C#### Apostrophe.
        IF (.NOT.F) THEN
          F=.TRUE.
          IF (I.EQ.0) GOTO 4
          GOTO 5
        ELSE
          IF (N.LE.LR .AND. REC(N:N).EQ.'''') THEN
            N=N+1
          ELSE
            F=.FALSE.
            IF (I.EQ.0) IA=IA+1
            GOTO 5
          ENDIF
        ENDIF
      ENDIF
C
      IF (I.LT.LA) THEN
        IF (I.EQ.0) IA=IA+1
        I=I+1
C#### Store character.
        AA(IA)(I:I)=C
      ENDIF
      GOTO 4
C
C#### Pad out with spaces.
5     IF (I.LT.LA) AA(IA)(I+1:)=' '
      IF (IA.LT.NA) GOTO 3
C
6     LDIGC=IA
      RETURN
C
C
      ENTRY LDIGT(NT,LP,MP)
C#### List-directed input, get token(s).
C#### NT = Number of token(s) required.
```

```
C#### LP = array of pointers to first character of token(s).
C#### MP = array of pointers to last character of token(s).
C#### Function value returned is the number tokens actually found in the
C#### current record (only 1 record is searched), or NT whichever is
C#### less.
C#### 1 or more spaces, tabs, comma, apostrophe & equals are delimiters.
C#### Unlike LDIGA, apostrophes may not appear in the input string.
C#### A comma following a delimiter skips the element.
C
      IT=0
      IF (NT.LE.0) GOTO 19
C
      F=.FALSE.
13    I=0
C
C#### Test for end-of-record.
14    IF (N.GT.LR) GOTO 19
C#### Get next char.
      C=REC(N:N)
      N=N+1
      IF (.NOT.F) THEN
        IF (C.EQ.' ' .OR. C.EQ.CHAR(9)) THEN
C#### Space or tab.
          IF (I.EQ.0) GOTO 14
          GOTO 18
C
        ELSEIF (C.EQ.',' .OR. C.EQ.'=') THEN
C#### Comma.
          IF (I.EQ.0) THEN
C#### Non-terminating comma or equals skips.
            IT=IT+1
            LP(IT)=N-1
          ENDIF
          GOTO 18
        ENDIF
      ENDIF
C
      IF ((F .AND. N.GT.LR) .OR. C.EQ.'''') THEN
C#### Apostrophe.
        IF (.NOT.F) THEN
          F=.TRUE.
          IF (I.EQ.0) GOTO 14
        ELSE
          F=.FALSE.
          IF (I.EQ.0) THEN
            IT=IT+1
            LP(IT)=N-1
          ENDIF
        ENDIF
        GOTO 18
      ENDIF
C
      IF (I.EQ.0) THEN
        IT=IT+1
        LP(IT)=N-1
        MP(IT)=N-2
      ENDIF
      I=I+1
      GOTO 14
C
18    MP(IT)=N-2
```

```
      IF (IT.LT.NT) GOTO 13
C
19    LDIGT=IT
      RETURN
C
C
      ENTRY LDIGI(NI,II)
C#### List-directed input, get integers.
C#### NI = Number of integer elements required.
C#### II = Array to receive NI integers.
C#### Function value returned is the number of elements actually found
C#### or NI, whichever is less.
C#### All characters except - and digits are delimiters.
C#### A comma following a delimiter skips the element.
C
      IA=0
      IF (NI.LE.0) GOTO 30
C
C#### Clear flag & set sign.
 21   F=.FALSE.
      IS=1
      N0=N
C
C#### Check for end-of-record.
22    IF (N.LE.LR) THEN
        C=REC(N:N)
        N=N+1
C
C#### Test for digit.
        I=INDEX(NUM,C)
        IF (I.EQ.0) THEN
          IF (C.EQ.'+') THEN
C#### Plus.
            IF (F) GOTO 28
          ELSEIF (C.EQ.'-') THEN
C#### Minus.
            IF (F) THEN
              N=N-1
              GOTO 28
            ENDIF
            IS=-1
          ELSE
            IF (C.EQ.',') THEN
C#### Comma.
              IA=IA+1
              GOTO 29
            ENDIF
            IF (F) GOTO 27
          ENDIF
C
        ELSE
          IF (.NOT.F) THEN
            F=.TRUE.
C#### Start integer element.
            IA=IA+1
            IH=ISIGN(I-1,IS)
          ELSE
C#### Accumulate.
            IH=10*IH+ISIGN(I-1,IS)
          ENDIF
        ENDIF
        GOTO 22
C
27      IF (C.NE.' ' .AND. C.NE.',' .AND. C.NE.CHAR(9)) THEN
          N=N0
          IF (F) IA=IA-1
          GOTO 30
        ENDIF
C
 28     II(IA) = IH
C#### Do we have enough.
 29     IF (IA.LT.NI) GOTO 21
      ENDIF
C
 30   LDIGI=IA
      RETURN
C
C
      ENTRY LDIGF(NF,FF)
C#### List-directed input, get floating.
C#### NF = Number of floating-point numbers required.
C#### FF = Array to receive nf floating point numbers.
C#### Function value returned is the number of numbers found,
C#### or NF whichever is less.
C#### Any character except - . E and digit will terminate.
C#### In E format (e.g. 1.234e-10) the "E" must not be separated from
C#### the mantissa (if this is omitted 1 is assumed).
C#### A comma following a delimiter skips the element.
C
      IA=0
      IF (NF.LE.0) GOTO 45
C
C#### Sign.
 31   IS=1
      N0=N
C#### Decimal point and exponent.
      IE=0
      JE=0
C#### Clear flags.
      F=.FALSE.
      FP=.FALSE.
      FE=.FALSE.
C
C#### Check for end-of-record.
32    IF (N.LE.LR) THEN
C#### Get next char.
        C=REC(N:N)
        N=N+1
C#### Test for digit.
        I=INDEX(NUM,C)
        IF (I.EQ.0) THEN
          IF (C.EQ.'+') THEN
C#### Test for end of number.
            IF (F) GOTO 43
          ELSEIF (C.EQ.'-') THEN
C#### Test for end of number.
            IF (F) GOTO 43
C#### Minus.
            IS=-1
          ELSEIF (C.EQ.'.') THEN
C#### Point.
            IF (FP .OR. FE) GOTO 42
            FP=.TRUE.
          ELSEIF (C.EQ.'E' .OR. C.EQ.'e') THEN
C#### E.
            IF (FE) GOTO 42
            FE=.TRUE.
C#### Test for number.
            IF (.NOT.F) THEN
C#### Supply 1.
              IA=IA+1
              DF=IS
            ENDIF
C#### Reset sign.
            IS=1
C#### Reset number flag.
            F=.FALSE.
C
          ELSE
C#### Test for number.
            IF (F .OR. FE) GOTO 42
            IF (C.EQ.',') THEN
C#### Comma.
              IA=IA+1
              GOTO 44
            ENDIF
            GOTO 42
          ENDIF
C
        ELSE
C#### Digit.
          IF (.NOT.FE) THEN
C#### Check for point.
            IF (FP) IE=IE-1
            IF (.NOT.F) THEN
C#### New number.
              F=.TRUE.
              IA=IA+1
              DF=ISIGN(I-1,IS)
            ELSE
C#### Accumulate.
              DF=10.*DF+ISIGN(I-1,IS)
            ENDIF
          ELSE
C#### Exponent.
            JE=10*JE+ISIGN(I-1,IS)
            F=.TRUE.
          ENDIF
        ENDIF
        GOTO 32
C
42      IF (C.NE.' ' .AND. C.NE.',' .AND. C.NE.CHAR(9)) THEN
          N=N0
          IF (F) IA=IA-1
          GOTO 45
        ENDIF
C
C#### Apply exponent.
43      IE=IE+JE
        IF (IE.GT.0) THEN
          DF=DF*10.**IE
        ELSEIF (IE.LT.0) THEN
          DF=DF/10.**(-IE)
        ENDIF
        IF (F) FF(IA)=DF
C         IF (F) WRITE(6,*) 'LDIGF',IA,FF(IA)
C
C#### Do we have enough.
44        IF (IA.LT.NF) GOTO 31
      ENDIF
C
 45   LDIGF=IA
      RETURN
C
C
      ENTRY LDIGP()
C#### List-directed input, get pointer to next byte in current record to
C     read.
C#### Calling routine should check if it is > record length.
50    IF (N.LE.LR .AND. (REC(N:N).EQ.' ' .OR. REC(N:N).EQ.CHAR(9))) THEN
        N=N+1
        GOTO 50
      ENDIF
      LDIGP=N
      END
C
C
      INTEGER FUNCTION LDIGH(NH,HH)
C#### List-directed input, get Hollerith(s).
C#### NH = Number of elements required (1 element = 4 chars max).
C#### HH = REAL array to receive NH elements.
C#### Function value returned is the number of elements actually found
C#### in the current record (only 1 record is searched), or NH whichever
C#### is less.
C#### Elements are returned left-aligned and upper-cased.
C#### 1 or more spaces, tabs, comma and apostrophe are delimiters.
C#### Apostrophes in the input string must be doubled and the whole
C#### string enclosed by apostrophes.
C#### A comma following a delimiter skips the element.
C
      IMPLICIT NONE
      INTEGER IH,NH
      CHARACTER A*4
      REAL HH(*)
      INTEGER LDIGA
C
      DO 1 IH = 1,NH
        IF (LDIGA(1,A).EQ.0) GOTO 2
        CALL CCPUPC(A)
 1      READ(A,'(A4)') HH(IH)
 2    LDIGH=IH-1
      END
C
C
      CHARACTER*(*) FUNCTION SUBSTR(A,L)
      IMPLICIT NONE
      CHARACTER A*(*)
      INTEGER L
C
      SUBSTR = A(:MIN(L,LEN(A)))
      END
```

ANNEX 6

```
      SUBROUTINE NAMELIST(NC,NKEY,KEYA,FMT,NFMT,TYPE,LDEF,HDEF,IDEF,
     &RDEF,LINP,LVAL,HVAL,IVAL,RVAL)
C
C#### Simulate namelist.
C
C#### Ian J. Tickle, Astex Technology.
C#### Copyright © 2000-2003 Astex Technology Ltd.  All rights reserved.
C
C#### This is steered completely by the imported variables NC & NKEY and
C#### by the imported arrays KEYA & FMT.
C
C#### Imported arguments:
C#### NC   = Minimum no of characters allowed in keyword.
C#### NKEY = No of keywords.
C#### KEYA = List of keywords, if blank defines >= 2nd array element.
C#### FMT  = List of formats for printing.
C
C#### Modified arguments:
C#### NFMT = List of number of items per keyword, if zero checks KEYA.
C#### TYPE = List of argument types ('L', 'H', 'I' or 'R') from FMT.
C#### LVAL = List of default LOGICAL values, input values exported.
C#### HVAL = List of default HOLLERITH values, input values exported.
C#### IVAL = List of default INTEGER values, input values exported.
C#### RVAL = List of default REAL values, input values exported.
C
C#### Note: LVAL, HVAL, IVAL & RVAL must be EQUIVALENCEd to each other
C#### and to COMMON block of input variables in calling program.
C
C#### Exported arguments:
C#### LDEF = Copy of imported LVAL:
C#### HDEF = Copy of imported HVAL:  must be EQUIVALENCEd.
C#### IDEF = Copy of imported IVAL:
C#### RDEF = Copy of imported RVAL:
C#### LINP = Flag to indicate keyword input.
C
      IMPLICIT NONE
C
      CHARACTER A*240, KEY*8
      LOGICAL P
      INTEGER I, IA, J, JA, K, KA, KE, LA, LC, LF, LK, MC, N, NC, NKEY
C
      CHARACTER FMT(*)*(*), KEYA(*)*(*), TYPE(*)
      LOGICAL LDEF(*), LINP(*), LVAL(*)
      INTEGER HDEF(*), HVAL(*), IDEF(*), IVAL(*), NFMT(*)
      REAL RDEF(*), RVAL(*)
C
      INTEGER LENSTR
C
      KE = 0
      DO I = 1,NKEY
        IF (KEYA(I).NE.' ') THEN
          IF (NFMT(I).EQ.0) THEN
            DO J = I+1,NKEY
              IF (KEYA(J).NE.' ') GOTO 13
              NFMT(I) = NFMT(I)+1
            ENDDO
          ENDIF
```

```
13        IF (TYPE(I).EQ.' ') THEN
            IF (INDEX(FMT(I),'L').GT.0 .OR. INDEX(FMT(I),'l').GT.0) THEN
              TYPE(I) = 'L'
            ELSEIF (INDEX(FMT(I),'A').GT.0 .OR. INDEX(FMT(I),'a').GT.0)
     &      THEN
              TYPE(I) = 'H'
            ELSEIF (INDEX(FMT(I),'I').GT.0 .OR. INDEX(FMT(I),'i').GT.0)
     &      THEN
              TYPE(I) = 'I'
            ELSEIF (INDEX(FMT(I),'E').GT.0 .OR. INDEX(FMT(I),'e').GT.0
     &      .OR. INDEX(FMT(I),'F').GT.0 .OR. INDEX(FMT(I),'f').GT.0 .OR.
     &      INDEX(FMT(I),'G').GT.0 .OR. INDEX(FMT(I),'g').GT.0) THEN
              TYPE(I) = 'R'
            ELSE
              WRITE(6,'(/A)') 'Bad FORMAT in NAMELIST.'
              TYPE(I) = 'R'
              KE = KE+1
            ENDIF
          ELSE
            CALL CCPUPC(TYPE(I))
          ENDIF
        ENDIF
        IDEF(I) = IVAL(I)
        LINP(I) = .FALSE.
      ENDDO
C
2     KA = -1
21    READ(*,'(A)',END=9) A
      LA = LENSTR(A)
      IF (LA.EQ.0) GOTO 21
      WRITE(6,'(2A)') 'Input line: ',A(:LA)
      I = INDEX(A(:LA),'#')
      IF (I.GT.0) THEN
        A(I:LA) = ' '
        LA = LENSTR(A(:LA))
        IF (LA.EQ.0) GOTO 21
      ENDIF
      IF (LA.GE.3 .AND. INDEX(A(:LA),'=').EQ.0) THEN
        CALL CCPUPC(A(:LA))
        IF ((LA.EQ.3 .OR. A(:MAX(LA-3,1)).EQ.' ') .AND.
     &  A(LA-2:LA).EQ.'END') GOTO 9
      ENDIF
C
3     DO IA = KA+2,LA
        IF (A(IA:IA).NE.' ') GOTO 5
      ENDDO
      GOTO 2
C
5     KA = IA+INDEX(A(IA:),',')-2
      IF (KA.EQ.IA-2) KA = LA
      IF (KA.LT.IA) THEN
        WRITE(6,'(/A)') 'Missing variable.'
        GOTO 8
      ENDIF
C
18    IF (KA.GT.IA .AND. A(KA:KA).EQ.' ') THEN
        KA = KA-1
        GOTO 18
      ENDIF
C
      JA = IA+INDEX(A(IA:KA),'=')-2
      IF (JA.EQ.IA-2) JA = KA
      IF (JA.LT.IA) THEN
        WRITE(6,'(/A)') 'Missing variable name.'
        GOTO 8
      ENDIF
C
      KEY = A(IA:JA)
      LK = LENSTR(KEY)
      CALL CCPUPC(KEY(:LK))
C      WRITE(6,'(3I5,2X,A)') IA,JA,LK,KEY(:LK)
C
      IA = JA+2
      IF (KA.LT.IA) THEN
        WRITE(6,'(/A)') 'Missing value for variable: '//KEY(:LK)
        GOTO 8
      ENDIF
C
19    IF (IA.LT.KA .AND. A(IA:IA).EQ.' ') THEN
        IA = IA+1
        GOTO 19
      ENDIF
C
      MC = MIN(LEN(KEY),LEN(KEYA(1)))
      LC = MIN(MAX(LK,NC),MC)
C      WRITE(6,'(3I5,2X,A)') IA,KA,LC,A(IA:KA)
C
      K = 0
      DO I = 1,NKEY
        IF (KEY(:LC).EQ.KEYA(I)(:LC)) THEN
          IF (K.GT.0) THEN
            WRITE(6,'(/A)') 'ERROR ambiguous variable name: '//KEY(:LK)
            GOTO 8
          ENDIF
          K = I
        ENDIF
      ENDDO
C
      IF (K.EQ.0) THEN
        WRITE(6,'(/A)') 'ERROR variable name not recognised: '//KEY(:LK)
        GOTO 8
      ENDIF
C
      N = 0
      DO I = IA+1,KA
        IF (A(I-1:I-1).NE.' ' .AND. A(I:I).EQ.' ') N = N+1
      ENDDO
C
      IF (NFMT(K).LT.0 .AND. N.GE.0) THEN
        NFMT(K) = N
      ELSEIF (N.NE.NFMT(K)) THEN
        WRITE(6,'(/A)') 'ERROR in value for variable: '//KEY(:LK)//
     &  '  "'//A(IA:KA)//'"'
        GOTO 8
      ENDIF
C
      IF (TYPE(K).EQ.'L') THEN
        READ(A(IA:KA),*,ERR=6) (LVAL(J),J=K,K+NFMT(K))
      ELSEIF (TYPE(K).EQ.'H') THEN
        CALL CCPUPC(A(IA:KA))
        DO J = K,K+NFMT(K)
          DO JA = IA+1,KA
```

```
            IF (A(JA:JA).EQ.' ') GOTO 1
          ENDDO
1         READ(A(IA:JA-1),'(A4)',ERR=6) HVAL(J)
          DO IA=JA+1,KA
            IF (A(IA:IA).NE.' ') GOTO 4
          ENDDO
4         CONTINUE
        ENDDO
      ELSEIF (TYPE(K).EQ.'I') THEN
        READ(A(IA:KA),*,ERR=6) (IVAL(J),J=K,K+NFMT(K))
      ELSEIF (TYPE(K).EQ.'R') THEN
        READ(A(IA:KA),*,ERR=6) (RVAL(J),J=K,K+NFMT(K))
      ELSE
        WRITE(6,'(/A)') 'Bad TYPE in NAMELIST.'
        TYPE(K) = 'R'
        KE = KE+1
      ENDIF
C
      DO J = K,K+NFMT(K)
        IF (LINP(J)) THEN
          WRITE(6,'(/A)') 'ERROR duplicate variable name: '//KEY(:LK)
          KE = KE+1
        ELSE
          LINP(J) = .TRUE.
        ENDIF
      ENDDO
      GOTO 3
C
6     WRITE(6,'(/A)') 'Invalid format for variable: '//KEY(:LK)//' = '//
     &A(IA:KA)
8     KE = KE+1
C
9     P = .TRUE.
      DO I = 1,NKEY
        IF (KEYA(I).NE.' ') THEN
          DO J = I,I+NFMT(I)
            IF (IVAL(I).NE.IDEF(I)) GOTO 10
          ENDDO
          IF (P) THEN
            WRITE(6,'(/A/)') 'Variables with default values:'
            P = .FALSE.
          ENDIF
          LF = LENSTR(FMT(I))
          LA = LF+9
          A(:LA) = '(T9,2A,'//FMT(I)(:LF)//')'
          IF (TYPE(I).EQ.'L') THEN
            WRITE(6,A(:LA)) KEYA(I),' =',(LVAL(J),J=I,I+NFMT(I))
          ELSEIF (TYPE(I).EQ.'H') THEN
            WRITE(6,A(:LA)) KEYA(I),' =',(HVAL(J),J=I,I+NFMT(I))
          ELSEIF (TYPE(I).EQ.'I') THEN
            WRITE(6,A(:LA)) KEYA(I),' =',(IVAL(J),J=I,I+NFMT(I))
          ELSE
            WRITE(6,A(:LA)) KEYA(I),' =',(RVAL(J),J=I,I+NFMT(I))
          ENDIF
        ENDIF
10      CONTINUE
      ENDDO
C
      KEY = ' '
      P = .TRUE.
      DO I = 1,NKEY
```

```
        IF (KEYA(I).NE.' ') THEN
          DO J = I,I+NFMT(I)
            IF (IVAL(I).NE.IDEF(I)) GOTO 16
          ENDDO
          GOTO 17
 16       IF (P) THEN
            WRITE(6,'(/A/)') 'Variables with non-default values:'
            P = .FALSE.
          ENDIF
          LF = LENSTR(FMT(I))
          LA = LF+8
          A(:LA) = '(T9,2A,'//FMT(I)(:LF)//')'
          IF (TYPE(I).EQ.'L') THEN
            WRITE(6,A(:LA)) KEYA(I),' !',(LDEF(J),J=I,I+NFMT(I))
            WRITE(6,A(:LA)) KEY(:MC),' =',(LVAL(J),J=I,I+NFMT(I))
          ELSEIF (TYPE(I).EQ.'H') THEN
            WRITE(6,A(:LA)) KEYA(I),' !',(HDEF(J),J=I,I+NFMT(I))
            WRITE(6,A(:LA)) KEY(:MC),' =',(HVAL(J),J=I,I+NFMT(I))
          ELSEIF (TYPE(I).EQ.'I') THEN
            WRITE(6,A(:LA)) KEYA(I),' !',(IDEF(J),J=I,I+NFMT(I))
            WRITE(6,A(:LA)) KEY(:MC),' =',(IVAL(J),J=I,I+NFMT(I))
          ELSE
            WRITE(6,A(:LA)) KEYA(I),' !',(RDEF(J),J=I,I+NFMT(I))
            WRITE(6,A(:LA)) KEY(:MC),' =',(RVAL(J),J=I,I+NFMT(I))
          ENDIF
 17       NFMT(I) = NFMT(I)+1
        ENDIF
      ENDDO
      IF (KE.GT.0) CALL CCPERR(1,'ERROR(S) in input.')
      END
```

SEQUENCE LISTING

SEQ ID No:1

```
ATGGCATACGGTACTCATTCACATGGTCTGTTTAAAAAACTGGGAATTCCAGGGCCCACACCTCTGCCTTTTTTGGG
AAATATTTTGTCCTACCATAAGGGCTTTTGTATGTTTGACATGGAATGTCATAAAAAGTATGGAAAAGTGTGGGGCT
TTTATGATGGTCAACAGCCTGTGCTGGCTATCACAGATCCTGACATGATCAAAACAGTGCTAGTGAAAGAATGTTAT
TCTGTCTTCACAAACCGGAGGCCTTTTGGTCCAGTGGGATTTATGAAAAGTGCCATCTCTATAGCTGAGGATGAAGA
ATGGAAGAGATTACGATCATTGCTGTCTCCAACCTTCACCAGTGGAAAACTCAAGGAGATGGTCCCTATCATTGCCC
AGTATGGAGATGTGTTGGTGAGAAATCTGAGGCGGGAAGCAGAGACAGGCAAGCCTGTCACCTTGAAAGACGTCTTT
GGGGCCTACAGCATGGATGTGATCACTAGCACATCATTTGGAGTGAACATCGACTCTCTCAACAATCCACAAGACCC
CTTTGTGGAAAACACCAAGAAGCTTTTAAGATTTGATTTTTTGGATCCATTCTTTCTCTCAATAACAGTCTTTCCAT
TCCTCATCCCAATTCTTGAAGTATTAAATATCTGTGTGTTTCCAAGAGAAGTTACAAATTTTTTAAGAAAATCTGTA
AAAAGGATGAAAGAAAGTCGCCTCGAAGATACACAAAAGCACCGAGTGGATTTCCTTCAGCTGATGATTGACTCTCA
GAATTCAAAAGAAACTGAGTCCCACAAAGCTCTGTCCGATCTGGAGCTCGTGGCCCAATCAATTATCTTTATTTTTG
CTGGCTATGAAACCACGAGCAGTGTTCTCTCCTTCATTATGTATGAACTGGCCACTCACCCTGATGTCCAGCAGAAA
CTGCAGGAGGAAATTGATGCAGTTTTACCCAATAAGGCACCACCCACCTATGATACTGTGCTACAGATGGAGTATCT
TGACATGGTGGTGAATGAAACGCTCAGATTATTCCCAATTGCTATGAGACTTGAGAGGGTCTGCAAAAAAGATGTTG
AGATCAATGGGATGTTCATTCCCAAAGGGGTGGTGGTGATGATTCCAAGCTATGCTCTTCACCGTGACCCAAAGTAC
TGGACAGAGCCTGAGAAGTTCCTCCCTGAAAGATTCAGCAAGAAGAACAAGGACAACATAGATCCTTACATATACAC
ACCCTTTGGAAGTGGACCCAGAAACTGCATTGGCATGAGGTTTGCTCTCATGAACATGAAACTTGCTCTAATCAGAG
TCCTTCAGAACTTCTCCTTCAAACCTTGTAAAGAAACACAGATCCCCCTGAAATTAAGCTTAGGAGGACTTCTTCAA
CCAGAAAAACCCGTTGTTCTAAAGGTTGAGTCAAGGGATGGCACCGTAAGTGGAGCCCACCATCACCATTGA
```

SEQ ID No: 2

```
MAYGTHSHGLFKKLGIPGPTPLPFLGNILSYHKGFCMFDMECHKKYGKVWGFYDGQQPVLAITDPDMIKTVLVKECY
SVFTNRRPFGPVGFMKSAISIAEDEEWKRLRSLLSPTFTSGKLKEMVPIIAQYGDVLVRNLRREAETGKPVTLKDVF
GAYSMDVITSTSFGVNIDSLNNPQDPFVENTKKLLRFDFLDPFFLSITVFPFLIPILEVLNICVFPREVTNFLRKSV
KRMKESRLEDTQKHRVDFLQLMIDSQNSKETESHKALSDLELVAQSIIFIFAGYETTSSVLSFIMYELATHPDVQQK
LQEEIDAVLPNKAPPTYDTVLQMEYLDMVVNETLRLFPIAMRLERVCKKDVEINGMFIPKGVVVMIPSYALHRDPKY
WTEPEKFLPERFSKKNKDNIDPYIYTPFGSGPRNCIGMRFALMNMKLALIRVLQNFSFKPCKETQIPLKLSLGGLLQ
PEKPVVLKVESRDGTVSGAHHHH
```

Further Description of the Invention

The invention is further described by the following numbered clauses:

1. A method of obtaining a representation of the three dimensional structure of a crystal of cytochrome P450 3A4, which method comprises providing the data of at least columns 1, 2, 3, 6 and 7 of Table 3 and constructing an electron density map of said data.

2. The method of clause 1 wherein said map is constructed by reference to the data of column 8 of said Table.

3. The method of clause 1 or 2 wherein an initial model of 3A4 is fitted to said map.

4. The method of clause 3 wherein said initial model is refined by reference to the data of columns 4 and 5 of said Table.

5. The method of any one of the preceding clauses which further comprises calculating the three-dimensional coordinates of one or more atoms of 3A4 in said crystal to provide a first three dimensional structure of 3A4.

6. The method of clause 5 wherein the structure is that of Table 5.

7. The method of clause 5 wherein the positions of one or more atoms in said first structure is varied to provide a second structure with three-dimensional coordinates having a r.m.s.d of less than 2.0 Å from said first structure.

8. A method of obtaining a representation of the three dimensional structure of a crystal of cytochrome P450 3A4, which method comprises providing the data of Table 5 or selected coordinates thereof, and constructing a three-dimensional structure representing said coordinates.

9. A computer-readable storage medium, comprising a data storage material encoded with computer readable data, the data comprising at least a selected portion of the three-dimensional coordinates of any one of clauses 5 to 8.

10. A computer-readable storage medium, comprising a data storage material encoded with computer readable data, wherein the data are defined by all or a portion of the structure coordinates of the P450 protein of Table 5 or a homologue of P450, wherein said homologue comprises backbone atoms that have a root mean square deviation from the backbone atoms of Table 5 of not more than 2.0 Å.

11. A computer-based method for the analysis of the interaction of a molecular structure with a P450 structure, which comprises:

providing the P450 structure obtainable by the method of any one of clauses 5 to 8 or selected coordinates thereof, the structures of any one of Table 5 or clauses 9 or 10 or selected coordinates thereof;

providing a molecular structure to be fitted to said P450 structure or selected coordinates thereof; and

fitting the molecular structure to said P450 structure.

12. The method of clause 11 wherein said selected coordinates include atoms from one or more of the residues of Phe57, Phe108, Phe213, Phe215, Phe219, Phe220, Phe241 and Phe304.

13. The method of clause 11 or 12 which further comprises the steps of:

obtaining or synthesising a compound which has said molecular structure; and

contacting said compound with P450 protein to determine the ability of said compound to interact with the P450.

14. The method of clause 11, 12 or 13 which further comprises the steps of:

obtaining or synthesising a compound which has said molecular structure;

forming a complex of a 3A4 P450 protein and said compound; and

analysing said complex by X-ray crystallography to determine the ability of said compound to interact with the P450.

15. The method of clause 14 which further comprises the steps of:

obtaining or synthesising a compound which has said molecular structure; and

determining or predicting how said compound is metabolised by said P450 structure; and

modifying the compound structure so as to alter the interaction between it and the P450.

16. A compound having the modified structure identified using the method of clause 15.

17. A method of obtaining an electron density map of a target P450 protein of unknown structure, the method comprises the steps of:

providing a crystal of said target P450;

obtaining an X-ray diffraction pattern of said crystal,

calculating an electron density map of said crystal by reference to the structure factor phase data of Table 3.

18. The method of clause 17 which further comprises modelling the structure of said target P450 of unknown structure on the 3A4 P450 structure obtainable by the method of any one of clauses 5 to 8 or the structures of any one of Table 5 or clauses 9 or 10 or selected coordinates thereof ; and

determining a conformation for said target P450 of unknown structure.

19. The method of clause 18 wherein said target P450 protein is selected from the group consisting of 3A5, 3A7 and 3A43.

20. A method for determining whether a compound is bound to P450 3A4 protein, said method comprising:

providing a crystal of said P450 protein;

soaking the crystal with the compound to form a complex; and

determining an electron density map of the complex by employing the data of Table 3 or a portion thereof.

21. The method of clause 20 which further comprises determining the structure of said compound.

22. The method of clause 20 which further comprises the steps of:

obtaining or synthesising the compound; and

modifying the compound structure so as to alter the interaction between it and the P450.

23. A computer system, intended to generate structures and/or perform optimisation of compounds which interact with P450, P450 homologues or analogues, complexes of P450 with

compounds, or complexes of P450 homologues or analogues with compounds, the system containing computer-readable data comprising one or more of:
(a) the structure factor data for P450 as shown in Table 3;
(b) atomic coordinate data obtainable by the method of clause any one of clauses 5 to 8 or defined by the structures of any one of Table 5 or clauses 9 or 10 or selected coordinates thereof, said data defining the three-dimensional structure of 3A4 P450 or at least selected coordinates thereof;
(c) atomic coordinate data of a target P450 protein generated by homology modelling of (b);
(d) atomic coordinate data of a target P450 protein generated by interpreting X-ray crystallographic data or NMR data by reference to the data of Table 3;
(e) structure factor data derivable from the atomic coordinate data of (c) or (d); and
(f) atomic coordinate data of Table 5 or a selected portion thereof.

24. A computer system according to clause 23, wherein said atomic coordinate data is for at least one of the atoms provided by the residues Phe57, Phe108, Phe213, Phe215, Phe219, Phe220, Phe241 and Phe304.

25. A computer system according to clause 23 or 24 comprising:
(i) a computer-readable data storage medium comprising data storage material encoded with said computer-readable data;
(ii) a working memory for storing instructions for processing said computer-readable data; and
(iii) a central-processing unit coupled to said working memory and to said computer-readable data storage medium for processing said computer-readable data and thereby generating structures and/or performing rational drug design.

26. A computer system according to clause 25 further comprising a display coupled to said central-processing unit for displaying said structures.

27. A method of providing data for generating structures and/or performing optimisation of compounds which interact with P450, P450 homologues or analogues, complexes of P450 with compounds, or complexes of P450 homologues or analogues with compounds, the method comprising:

(i) establishing communication with a remote device containing computer-readable data comprising at least one of: (a) the structure factor data for P450 as shown in Table 3; (b) atomic coordinate data obtainable by the method of any one of clauses 5 to 7 or defined by the structures of any one of Table 5 or clauses 9 or 10 or selected coordinates thereof;, said data defining the three-dimensional structure of P450, or selected coordinates of atoms of P450; (c) atomic coordinate data of a target P450 homologue or analogue generated by homology modelling of the target based on the data (b); d) atomic coordinate data of a protein generated by interpreting X-ray crystallographic data or NMR data by reference to the data of Table 3; and (e) structure factor data derivable from the atomic coordinate data of (c) or (d); and

(ii) receiving said computer-readable data from said remote device.

28. The method of clause 27 wherein said atomic coordinate data is that of Table 5 or a selected portion thereof.

29. A computer-readable storage medium comprising a data storage material encoded with computer-readable data, wherein the data are defined by:

(a) the structure factor data for P450 as shown in Table 3;

(b) atomic coordinate data obtainable by the method of any one of clauses 5 to 7 or defined by the structures of any one of Table 5 or clauses 9 or 10 or selected coordinates thereof;, said data defining the three-dimensional structure of 3A4 P450 or at least selected coordinates thereof;

(c) atomic coordinate data of a target P450 protein generated by homology modelling of the target based on the data of (b);

(d) atomic coordinate data of a target P450 protein generated by interpreting X-ray crystallographic data or NMR data by reference to the data of Table 3;

(e) structure factor data derivable from the atomic coordinate data of (c) or (d); and
(f) atomic coordinate data of Table 5 or a selected portion thereof.

30. A computer-readable storage medium comprising a data storage material encoded with a first set of computer-readable data comprising a Fourier transform of at least a portion of the structural coordinates for the P450 protein obtainable by the method of any one of clauses 5 to 7 or defined by the structures of any one of Table 5 or clauses 9 or 10 or selected coordinates thereof;; which data, when combined with a second set of machine readable data comprising an X-ray diffraction pattern of a molecule or molecular complex of unknown structure, using a machine programmed with the instructions for using said first set of data and said second set of

data, can determine at least a portion of the structure coordinates corresponding to the second set of machine readable data.

31. The computer-readable storage medium of clause 30 wherein said first set of computer-readable data comprise a Fourier transform of at least a portion of the structural coordinates for the P450 protein of Table 5 or selected coordinates thereof.

32. A method of determining an electron density map of a target protein which is, or is homologous to, 3A4, which method comprises providing a crystal of the target protein, obtaining an X-ray diffraction of said protein, and generating an electron density map of said target protein by reference to the structure factor phase data of Table 3.

33. A crystal of P450 3A4 protein having a resolution better than 3.1 Å.

34. A crystal of P450 protein having the structure defined by the co-ordinates of Table 5.

36. A method of predicting three dimensional structures of P450 homologues or analogues of unknown structure, the method comprises the steps of:

aligning a representation of an amino acid sequence of a target P450 protein of unknown three-dimensional structure with the amino acid sequence of the P450 of Table 5 to match homologous regions of the amino acid sequences;

modelling the structure of the matched homologous regions of said target P450 of unknown structure on the corresponding regions of the P450 structure as defined by Table 5; and

determining a conformation for said target P450 of unknown structure which substantially preserves the structure of said matched homologous regions.

37. The method of clause 36 wherein said target P450 protein is selected from the group consisting of 3A5, 3A7 or 3A43.

38. A chimaeric protein having a binding cavity which provides a substrate specificity substantially identical to that of P450 3A4 protein,

wherein the chimaeric protein binding cavity is lined by a plurality of atoms which correspond to selected P450 3A4 atoms lining the P450 3A4 binding cavity, the relative positions of said plurality of atoms corresponding to the relative positions, as defined by Table 5, of said selected P450 3A4 atoms.

39. A method for determining the structure of a protein, which method comprises;

providing the co-ordinates of Table 5 or selected coordinates thereof, and

either (a) positioning said co-ordinates in the crystal unit cell of said protein so as to provide a structure for said protein, or (b) assigning NMR spectra peaks of said protein by manipulating said co-ordinates.

40. A method for determining the structure of a compound bound to P450 protein, said method comprising:

providing a crystal of P450 protein;

soaking the crystal with the compound to form a complex; and

determining the structure of the complex by employing the data of Table 5 or a portion thereof.

41. A method for determining the structure of a compound bound to P450 protein, said method comprising:

mixing P450 protein with the compound;

crystallizing a P450 protein-compound complex; and

determining the structure of the complex by employing the data of Table 5 or a portion thereof.

42. A method of assessing the ability of a compound to interact with P450 3A4 protein which comprises:

obtaining or synthesising said compound;

forming a crystallised complex of a P450 3A4 protein and said compound, said complex diffracting X-rays for the determination of atomic coordinates of said complex to a resolution of better than 2.8 Å; and

analysing said complex by X-ray crystallography to determine the ability of said compound to interact with the P450 3A4 protein.

43. Use of the atomic coordinate data or selected coordinates thereof of Table 5 for the provision of a computer-generated structure of a cytochrome P450 molecule bound to a ligand.

44. A computer-based method of rational drug design comprising:

(a) providing the coordinates of at least two atoms of a P450 3A4 structure as defined in Table 5 ± a root mean square deviation from the Cα atoms of less than 1.5 Å ("selected coordinates");

(b) providing the structures of a plurality of molecular fragments;

(c) fitting the structure of each of the molecular fragments to the selected coordinates; and (d) assembling the molecular fragments into a single molecule to form a candidate modulator molecule.

45. The method of clause 44 further comprising the step of:

(a) obtaining or synthesising the molecular structure or modulator; and

(b) contacting the molecular structure or modulator with P450 3A4 to determine the ability of the molecular structure or modulator to interact with P450 3A4.

46. A method for identifying a candidate modulator of P450 3A4 comprising the steps of:

(a) employing a three-dimensional structure of P450 3A4, at least one sub-domain thereof, or a plurality of atoms thereof, to characterise at least one P450 3A4 binding cavity, the three-dimensional structure being defined by atomic coordinate data according to Table 5 ± a root mean square deviation from the Cα atoms of less than 1.5 Å; and

(b) identifying the candidate modulator by designing or selecting a compound for interaction with the binding cavity.

47. The method of clause 46 further comprising the step of:

(a) obtaining or synthesising the molecular structure or modulator; and

(b) contacting the molecular structure or modulator with P450 3A4 to determine the ability of the molecular structure or modulator to interact with P450 3A4.

All publications and patents mentioned in the above specification are herein incorporated by reference. Various modifications and variations of the described invention will be apparent to those of skill in the art without departing from the scope and spirit of the invention. Although the invention has been described in connection with specific preferred embodiments, it should be understood that the invention as claimed should not be unduly limited to such specific embodiments.